PROKARYOTIC GENE START PREDICTION: ALGORITHMS FOR GENOMES AND METAGENOMES

A Dissertation
Presented to
The Academic Faculty

By

Karl Gemayel

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Computational Science and Engineering

Georgia Institute of Technology

December 2020

PROKARYOTIC GENE START PREDICTION:
ALGORITHMS FOR GENOMES AND METAGENOMES

Thesis committee:

Dr. Mark Borodovsky
School of Computational Science and Engineering and Department of Biomedical Engineering
Georgia Institute of Technology

Dr. Polo Chau
School of Computational Science and Engineering
Georgia Institute of Technology

Dr. Ümit Çatalyürek
School of Computational Science and Engineering
Georgia Institute of Technology

Dr. King Jordan
School of Biological Sciences
Georgia Institute of Technology

Dr. Pen Qui
Department of Biomedical Engineering
Georgia Institute of Technology

Date approved: October 31, 2020

Wooster: "There are moments, Jeeves, when one asks oneself, 'Do trousers matter?'"

Jeeves: "The mood will pass, sir."

P.G. Wodehouse, The Code Of The Woosters

To Mom and Dad,

all my ancestors, and the

first self-replicating molecule.

Without you, this work would literally

not have been possible.

ACKNOWLEDGMENTS

I am bound to forget someone or something and so, in fairness to all, I will forget most things and keep this vague and terse, though not necessarily short.

I was very much at the right place at the right time to do this work, a time where these problems had not yet been solved. To the driven students who graduated early enough before such ideas came to them, thank you for being considerate.

To my advisor Mark Borodovsky, who insisted that a lack of community funding for prokaryotic gene finding does *not* mean that the problem has actually been solved, thank you for continuously pushing for rigorous science that questions accepted beliefs. To my committee members Umit Catalyurek, Polo Chau, King Jordan, and Peng Qiu, whose questions and perspectives pushed me to think of my work in a much larger context than I would otherwise have done, thank you all showing me the need and benefit of bursting the bubble every once in a while.

To Richard Fujimoto, who introduced me to teaching and allowed me to do it my way. You were a continuous source of kindness, support, and encouragement. To Charles Isbell, who let me teach one of the most popular CS courses. Not once, but thrice. *Why?* To the faculty members who've always been kind (Polo, Rich, Umit, Bistra, and Richard again), and to those who've not been unkind. To Umit, I could write an article... To my undergraduate advisor, Fatima Abu Salem, whose early and frankly heartwarming encouragement first set me on my path to a life of research; much of the early credit, with some of the blame, goes to you.

To all my friends, let me preface by assuring you that "*et cetera*", abbreviated *etc*, is a neutral, non-diminishing term used to describe similar items not *necessarily* of lower quality than those previously listed. With that in mind, and in shuffled order, thank you to Sami, Lyes, Tomáš, Amrita, Anant, Fadi, Benito, "*et cetera*."

To Chloe and Kirk, who wisely joined towards the end of my degree. Thank you for

the snarkiness and irony that easily rivaled mine, and for the eccentricity that I could only aspire to. I'll let you both figure out who's who.

To my elder brothers Nader and Roland, for getting their doctorates before I got mine, forging a deceptively clear path. To my younger brother Peter, for not getting his before I got mine.

To my parents Nina and Michel, for their continuous support and push towards a good education, and because I just got my PhD and I'm on my way to unemployment; I may need a place to stay. I am also sorry for skipping my bachelors and masters graduation ceremonies with the promise that we will all attend "the big one". This time, I have an actual excuse. Thanks Covid.

To other family members, who I gladly and lovingly consider my friends. Please see my comment on "*et cetera*."

To Mia, for providing the necessary support and chocolate in difficult times, for subsequently taking away said chocolate when necessary, and for pushing me to continue down this path while simultaneously and unapologetically showing me that a masters degree would have been enough.

Vague and long... To my undergraduate students who've taught me that there is *never* a fear of not meeting the 10 page report requirement, I represent your spirit here. Still, I wish you had all realized that 10 pages was the *upper limit*.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

dRNA-seq differential RNA sequencing

HMM hidden Markov model

LORF longest open-reading-frame. *Glossary:* Longest ORF

LOWESS locally weighted scatterplot smoothing

mRNA messenger RNA

RBS ribosomal binding site. *Glossary:* Ribosomal binding site

SD-RBS Shine-Dalgarno RBS. *Glossary:* Shine-Dalgarno RBS

UTR untranslated region

LIST OF TERMS

16S rRNA component of the ribosome that binds to the RBS during translation-initiation.

Leaderless transcription A mode of transcription where the transcription start site is very close to or at the start of the first gene in the transcript. Its implication here is that no RBS can exist upstream of that gene.

Longest ORF The longest possible ORF in a sequence, acquired by extending an ORF to the upstream-most possible start codon, e.g. ATG, before reaching a possible stop codon, e.g. TAA. *Glossary:* Open-reading-frame

Open-reading-frame A part of a sequence that has the ability to be translated into a protein.

Promoter A DNA-level signal (usually 6nt long in bacteria) that helps identify the transcription start site.

Ribosomal binding site An mRNA-level signal (usually 6nt long in bacteria) used by the ribosome to identify the start of a gene.

Shine-Dalgarno RBS A type of RBS whose consensus has the form AGGAGG. This is the most common type found in prokaryotes.

Translation-initiation The biological mechanism that aids in detecting the start of a gene, following which gene-to-protein translation can occur.

SUMMARY

Prokaryotic gene-prediction is the task of finding genes in archaeal or bacterial DNA sequences. These genomes consist of alternating gene-coding and non-coding regions, meaning the task is solved by determining the start and end points of each gene in the DNA sequence, with gene-start prediction generally considered to be more difficult. The primary focus of this work is to improve gene-start prediction accuracy and our understanding of the biological translation-initiation mechanisms used to mark and determine gene-starts.

There are two challenges that characterize this task. First, ground-truth, experimentally verified gene-starts are only available for a very small set of genes, and second, our knowledge of translation-initiation mechanisms is incomplete and quite often misleading. Three motivating questions arise from these challenges and are addressed in this work.

First, how can we predict gene-starts in a DNA sequence without relying on ground-truth data and without any prior biological knowledge of that species? I show how simplifying assumptions about translation-initiation mechanisms biased the design of existing gene-finder algorithms hindering their predictive performance. I present GeneMarkS-2, an algorithm that relaxes those assumptions and learns more accurate representations of these mechanisms, thereby achieving more accurate predictions. Using it, I provide an updated view of the diversity of translation-initiation mechanisms across the prokaryotic domain. GeneMarkS-2 is now used by the National Center for Biotechnology Information (NCBI) to annotate their database of more than two hundred thousand prokaryotic genomes.

Second, how can we measure the accuracy of gene-start prediction without access to ground-truth data? I show that the accuracy of existing methods measured on the limited set of verified data does not generalize to the much larger and more diverse set of available genes. This proves that these benchmark sets of verified starts are not representative enough for this task. I describe an alternative method to boost prediction performance for genes outside the ground-truth set by effectively filtering low-certainty predictions. This is

done by only selecting gene-start predictions that are corroborated by multiple, independent sources of evidence. As part of this approach, I propose StartLink, a new comparative genomics approach for gene-start prediction; that is, comparing DNA fragments from multiple species rather than relying solely on a single genome.

Third, how can we predict gene-starts for metagenomes, i.e. cases where frequently only part of the DNA sequence is available? Here, I describe how the mechanisms for gene-start prediction developed for GeneMarkS-2 can be ported to metagenomes, which often have short DNA fragments that hinder the performance of predictive methods. I present MetaGeneMarkS, and show that it achieves accuracies on metagenomes close to those achieved by GeneMarkS-2 on fully-sequenced DNA.

Several recurring themes appear throughout this work. Understanding the limits of our knowledge of translation-initiation mechanisms proves essential to designing better models and provides an open field of new exploration of the diversity of these mechanisms. Furthermore, our unhealthy dependence on verified gene-starts for measuring performance has and continues to prevent us from accurately portraying the quality of our predictors, despite the $>95\%$ average accuracy levels measured on this set. It is therefore critical to restate that gene-start prediction is still an open problem.

CHAPTER 1

INTRODUCTION

Over the course of over three decades, research on gene prediction has made significant progress. However, strong assumptions of the dominant biological mechanisms used in identifying gene-starts resulted in flawed modeling in existing gene-start prediction tools, leading to less accurate predictions. This bias has helped paint an inaccurate picture of these mechanisms and their prevalence in the prokaryotic domain.

There are two common approaches to predicting genes and their starts: ***ab initio*** and ***comparative genomics***. An *ab initio* method relies solely on the DNA of a given species, and builds a "native" representation specific to that species' DNA without the use of external evidence. This representation is then used to identify genes within the DNA sequence. In contrast, comparative genomics makes use of the evolutionary-based relationship between different species. It uses the similarities and differences between DNA sequences of multiple species to identify genes, under the assumption that gene-coding regions are less likely to mutate and differ than non-coding regions.

Both *ab initio* and comparative methods are developed and used throughout this work to overcome previously unaddressed challenges in gene-start prediction. Specifically, this thesis is organized as follows. **Chapter 2**[1] describes the biological background needed to properly and comprehensively formulate the problem of gene and gene-start prediction. This is followed by three chapters that describe new developments.

Chapter 3: Gene prediction on complete genomes: Given the genome (full DNA sequence) of an unknown species, our objective is to find the locations of genes in that sequence. In this chapter, I describe GeneMarkS-2, a new algorithm that uses unsupervised

[1] Underlined words represent "clickable" references to other parts of this text. This includes references to sections, figures, equations, as well as acronym and glossary entries.

learning to build a model of the given genome and uses it to find genes in the sequence. I show how it achieves better accuracies due to its more complex gene-start model. Using GeneMarkS-2, I also show a new picture of the diversity of translation-initiation mechanisms in archaea and bacteria.

Chapter 4: Improving generalization in the absence of ground-truth data: The biggest challenge in gene-start prediction is the very limited set of ground-truth data, i.e. genomes with experimentally verified gene-starts. This makes designing algorithms challenging and, more importantly, makes it impossible to accurately validate these algorithms. This work shows that the performance of existing tools on the available ground-truth data does not generalize to the much wider set of unlabeled data. In light of this, I present an approach to increase our confidence in predictions by combining independent sources of evidence; this removes uncertain predictions, dramatically increasing the reliability of those that remain behind.

Chapter 5: Gene prediction on metagenomes: Contrary to the case described in Chapter 3, we are often presented with a short fragment of DNA sequence rather than a complete one. This occurs when the DNA of a single species cannot be fully sequenced, such as in metagenomic samples. In this case, trying to learn parameters from a single short sequence becomes a challenge. I describe MetaGeneMarkS, an algorithm that learns a "meta-model" from a large set of pre-trained GeneMarkS-2 models. This can then be used to find genes in short and long sequences, without the need to train sequence-specific parameters.

Chapter 6 concludes this work by summarizing key biological and algorithm insights, describes some of the open problems that remain, and suggests possible steps that might help answer some of these questions.

CHAPTER 2
BACKGROUND

This chapter briefly describes important concepts that will set the stage for Chapters 3 to 5. I discuss the biology behind DNA, genes, and gene-starts, and the mechanisms for gene-start detection. I also describe methods to experimentally determine reliable gene-starts, which are a crucial and controversial basis for benchmarking algorithm performance. Finally, I describe the general framework of gene-finding algorithms.

2.1 What's in a gene?

A basic understanding of the relevant parts of DNA structure and the mechanisms necessary to convert genes into proteins are crucial to understanding the intuition and justification behind many of the design decisions in this work.

The biological process of going from a gene to the corresponding protein can be broken up into two high-level steps. First, a segment of DNA containing one or more genes is copied to a messenger RNA (mRNA) through a process called **transcription**. Then, a molecule called the **ribosome** converts genes in this segment to their corresponding amino-acid sequences through the process of **translation** (Figure 2.1). Both these processes use information that helps the cell identify where genes and their starts are located.

2.1.1 Transcription

Simply stated, transcription is the process of making an exact copy of a contiguous DNA fragment (called an **operon**) that contains one or more genes. First, a molecule called RNA polymerase binds to a fragment of the DNA called the promoter. The promoter consists of two short sequences positioned at about -10 and -35 nucleotides (*nt*) upstream of the start

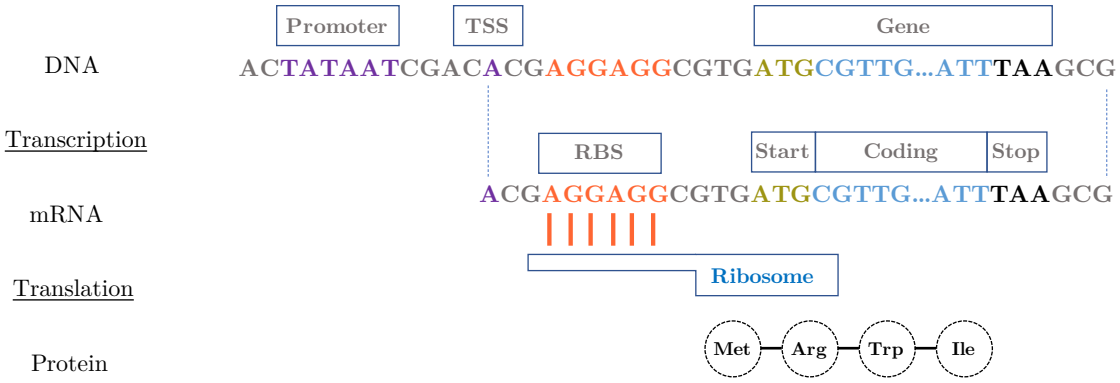


Figure 2.1: Illustration of transcription and translation processes in prokaryotic genomes.

of the transcription starting point.[1] The "-" means that the promoter is located upstream, or "before," the gene-start. Both sequences usually have a length of 6 nucleotides. The -10 element is called the Pribnow box, and is essential to start transcription in prokaryotes, while the -35 element, if it exists, can lead to increased transcription rates. Figure 2.1 shows an example of a promoter.

Once RNA polymerase binds to the promoter, transcription of a nearby fragment begins, and a region containing one or more genes is copied to an mRNA. It is worth noting that the copied fragment does not only contain genes, but also other signals that can help with the translation process that converts the genes into proteins.

2.1.2 Translation

At a high level, translation follows steps similar to transcription. First, a ribosome molecule simultaneously binds to two regions:

1. A fragment called the RBS, around 6 *nt* in length, that exists at roughly -7 *nt* upstream of the start of a gene. The typical consensus sequence (i.e. its most common form) of an RBS is AGGAGG, also referred to as the Shine-Dalgarno sequence [1].

2. The start region of the gene, typically containing either ATG, GTG, or TTG.

[1] Upstream and downstream positions refer to regions "before" and "after" a particular point in the DNA.

Once binding occurs, the ribosome then proceeds to begin decoding the gene into its corresponding protein sequence, moving one **codon** (i.e. 3 *nt*) at a time, until it reaches the end of the gene. A gene typically ends at the first TAA, TGA, or TAG codon.

2.1.3 Ribosomal Binding Sites (RBS) and 16S rRNA

Ribosomal binding sites help the ribosome determine the start of a gene. Studies have demonstrated that disrupting these sites through targeted mutations also disrupts the translation of genes into proteins. It is therefore no surprise that modeling these sites correctly helps in gene-start prediction.

2.1.3.1 Shine-Dalgarno RBS

The most common type of RBS is the Shine Dalgarno RBS (SD-RBS), whose most common form is the AGGAGG motif, or some sub-sequence of it. The frequency of these motifs depends on the type of genome, including its GC-content. This is possibly due to rich G content of this motif, and that G nucleotides are more frequent in high GC genomes which can lead to random RBS-looking signals.

At translation-initiation, the tail of the 16S rRNA, a component of the ribosome, binds to the RBS which serves as an indicator that the gene-start is nearby (Figure 2.1). This binding allows the ribosome to initiate translation, and also reduces secondary structure folding in the mRNA, leaving the gene-initiation area exposed for initiation [2, 3].

2.1.3.2 RBS motifs with a non-Shine Dalgarno consensus

More recently, attention has been directed to genomes where some genes do not have a matching SD-RBS located upstream of the gene-start. For example, some non-SD motifs were found in E. coli [3, 4], but these verified instances were not directly mapped to other genomes. In many instances, computational studies detected large percentages of genes that lacked an SD-RBS, but they could not always come to definitive conclusion whether that

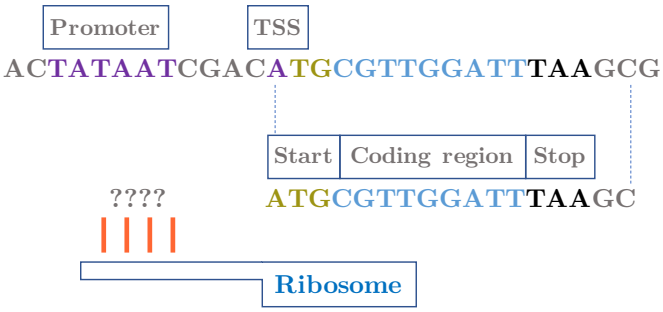


Figure 2.2: An illustration of leaderless transcription

was due to a non-SD RBS, a non-RBS mechanism, or even incorrect gene-start annotation. These were often done with a direct sequence matching of the AGGAGG consensus to the upstream sequences [5].

Non-SD RBS typically have an A-rich structure, such as TAAAAA. They provide the same translation-initiation benefits as SD-RBS, and mutating them reduces or halts the translation of that gene.

The presence of non-SD RBS has not been fully explored with respect to translation efficiency. Previous studies have shown that the ribosomal protein S1 can bind to non-SD motifs to mediate translation initiation in E. coli [3, 4]. However, the S1 protein does not appear to be essential in many other prokaryotes [5, 6], suggesting that there may be other mechanisms at play.

2.1.4 Leaderless transcription

While the above description generally holds, there are many instances in which the start of transcription is extremely close to or at the start of the gene. In other words, the gene begins at the start of the mRNA, leaving no room for an RBS. In this case, the typical RBS model breaks down (Figure 2.2).

This process is called leaderless transcription, referring to a gene in the mRNA without an upstream sequence. It was previously shown to be common in archaea and, generally, rare in bacteria. As such, it was largely ignored in gene-prediction tools.

The fraction of genes with leaderless transcription was observed to vary significantly among species [7]. It is low in some bacteria (<8% among all operons) such as *Helicobacter pylori* [8], *Bacillus subtilis* [9], *Salmonella enterica* [10], *Bacillus licheniformis* [11], *Campylobacter jejuni* [12], *Propionibacterium acnes* [13], *Shewanella oneidensis* [14], and *Escherichia coli* [15]. It is also low in some archaea (<15%), e.g., in *Methanosarcina mazei* [16], *Pyrococcus abyssi* [17], *Thermococcus kodakarensis* [18], *Methanolobus psychrophilus* [19], and *Thermococcus onnurineus* [20]. However, a higher frequency (>25%) of leaderless transcription was observed in other bacteria, e.g., *Mycobacterium tuberculosis* [7], *Corynebacterium glutamicum* [21], *Deinococcus deserti* [22], *Streptomyces coelicolor* [23], *Mycobacterium smegmatis* [24], and an even larger frequency (>60%) was seen in various archaeal species, e.g., Halobacterium salinarum [25], Sulfolobus solfataricus [26], and Haloferax volcanii [6]. Accordingly, the diversity of regulatory sequence patterns that appear near gene starts motivated the effort to build multiple models necessary for more accurate gene-start prediction.

Our understanding of leaderless transcription in bacteria has grown rapidly in the last two decades. For example, leaderless transcription was previously thought to be very rare in bacteria; in 2002, one study found that after a decade of sequencing, only 35-40 leaderless mRNA transcripts were found. We now know of more genomes in which leaderless transcription is common. However, attempts to gather all the works done on individual genomes into a comprehensive picture were few and far between, leading to an incorrect view of the diversity of gene initiation mechanisms in the bacterial world.

2.2 N-terminal sequencing: The bronzed gold-standard for verified gene-starts

N-terminal sequencing is a chemical approach to determining short fragments of a protein, specifically those near the start (N-terminus). These fragments can be mapped back to the DNA sequence to determine gene-starts. While the details of this process are beyond the scope of this work, I will highlight some of the characteristics of gene-start data sets

determined by N-terminal sequencing.

There are very few genes with starts verified by N-terminal sequencing. This is partly because N-terminal sequencing has been replaced by mass spectrometry in many applications, which is more effective at tasks such as protein identification and characterization [27]. This has led to a decrease in N-terminal experiments. Unfortunately, mass spectrometry does not provide the same resolution for gene-start labels.

For genomes where N-terminal sequencing data is available, it is not available for all genes within these genomes. Specifically, N-terminal sequencing requires that large quantities of a protein are produced, which is not the case for all genes. Furthermore, some proteins are blocked through post-translation modifications to the N-terminus, and therefore cannot be sequenced with this approach.

The combination of these factors has led to a very small and unrepresentative set of experimentally verified gene-starts. Nevertheless, this set has been used in many studies to measure the accuracy of gene-start predictions. My work shows that, while useful, this set is not enough to accurately benchmark algorithms for gene-start prediction.

2.3 *Ab initio* versus comparative genomics

Consider the history of the DNA of a species of interest. Most fragments of this DNA have existed in this sequence for quite a while (Figure 2.3). They were passed on through generations, from parent to child, and have been exposed to similar environmental forces, which leads them to have similar properties. For example, environmental forces have been shown to affect the frequency of G and C (compared to A and T) nucleotides in a genome [28]. Therefore, learning about a particular genome as a whole has some benefits, including the ability to model compositional properties that are preserved across the genome.

Through its evolution, the information in this DNA sequence has also generated many sisters, aunts, and first cousins thrice removed (Figure 2.3). While these relatives can differ in many ways, they still share a many properties. Understanding that history can help

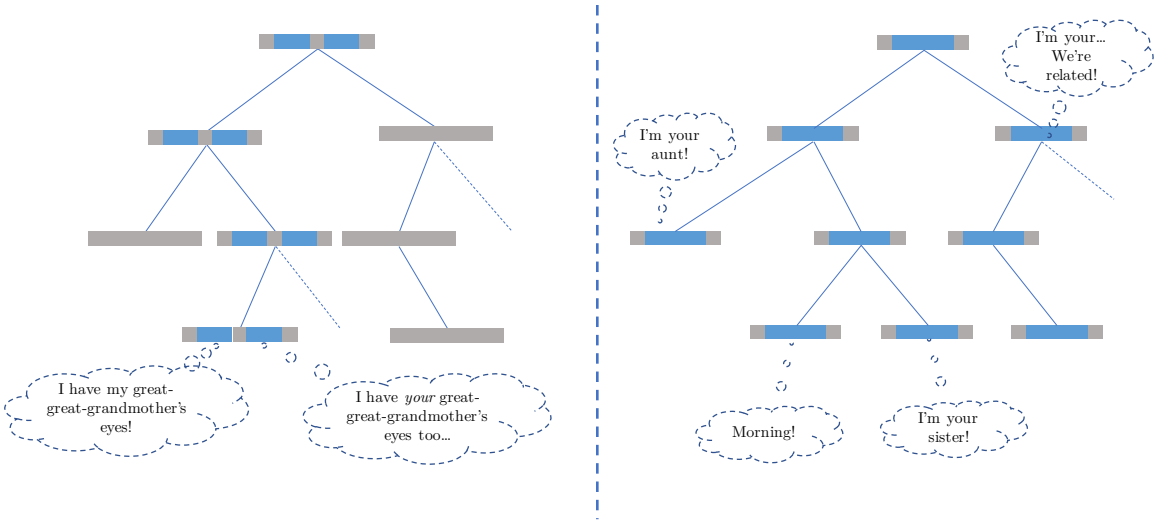


Figure 2.3: The conservation of properties of genes with a similar ancestry (left), and the similarity across different lineages with the same ancestor (right).

identify important pieces of information that have been conserved or lost over time.

The above two world views lead to two different approaches of gene prediction: *ab initio* and comparative analysis. *Ab initio* gene-prediction methods rely mainly on the information found in a specific genome. They analyze a single genomic sequence and determine native properties (e.g. RBS model, gene model, etc.) that can be used to find genes. On the other hand, comparative analysis relies on detecting similarities and differences between related species. Under the assumption that "useful" information such as genes are more likely to be preserved than non-coding regions, this approach can look for conserved regions across distant relatives and use that information to predict genes.

2.3.1 Pros and Cons

The most obvious benefit of *ab initio* algorithms is that they do not rely on existing databases. This is critical when trying to find genes that have not been found in previously tested species. Since the number of microbial species on Earth is estimated to be larger than 10^{12} [29], and given that we have sequenced less than 0.00002% of that, many genes are still left to be discovered. In such cases, *ab initio* methods are our only way forward.

On the other hand, comparative methods allow us to use the ever-increasing amount of

biological data that is being gathered. In particular, such approaches can filter out gene-like fragments that may randomly appear as genes in a single genome. They are also useful in transferring known functions of a given protein from one species to a sister gene in another species (though this is beyond the scope of this thesis).

Currently, pipelines such as NCBI's PGAP, used to create the RefSeq database, rely on a combination of *ab initio* and comparative approaches. In fact, at the time of this writing, PGAP uses GeneMarkS-2 (see Chapter 3) as its *ab initio* component.

2.4 Why is gene start prediction useful?

The correct identification of gene-starts can lead to a greater understanding of evolutionary relationships between species and their translation-initiation mechanisms.

For example, a 2011 study showed that the change of translation initiation mechanisms (e.g. RBS, leaderless transcription, etc...) is linearly dependent on the phylogenetic relationship between species [4]. It also suggested that leaderless transcription can provide a better understanding of the ancestral relationship between archaea and bacteria, two of the highest domains in the kingdom of living things.

Furthermore, correct gene-start identification is necessary to study the biological and biochemical properties of gene-starts and their upstream regions. These regions can have an effect on the conditions and frequencies at which genes are translated into proteins, through translation- and transcription-based gene regulation [30–32].

For example, bacteria must be able to adjust to changing forces in their environments, and the monitoring of these forces and their response to them are related to the frequency of gene translation. It also is known that ribosomal binding sites play a role in a gene's translation-initiation rate [30]. Therefore, the exact identification of gene-start regions, and thus RBS sites, provides a step forward in understanding regulation by gene translation. This can be done, for example, by mutating regions around gene-starts and observing the consequent behavior. It is also shown that secondary structures formed by the ribosomal

binding site can be used to inhibit translation [30]. For example, this mechanism is used by the cell to quickly respond to heat shocks, i.e. quick rises in temperature.

Correct identification of gene-starts also allows us to better understand the diversity of translation-initiation mechanisms. For example, as described in Chapter 3, the frequency of leaderless transcription in bacteria was greatly underestimated in previous works. As shown later, better modeling of these mechanisms has led to a dramatic shift in our understanding of the diversity of translation-initiation mechanisms in prokaryotes.

From a more practical perspective, gene-start identification can be important to lab-induced changes in gene-expression. For example, [33] showed that some antibiotics inhibit translation initiation on leadered transcripts, but not on leaderless transcripts. Therefore, the correct identification of gene-starts, which can in turn identify leaderless transcripts, allows further study in that space.

The wide range of experiments that would benefit from better gene-start identification is endless. This should not come as a surprise, given that gene transcription and translation, along with the generated proteins, form the not-yet-fully-understood building blocks of molecular biology.

CHAPTER 3

GENE START PREDICTION USING GENEMARKS-2

As the rate at which new DNA is sequenced continues to increase, *ab initio* gene-finding algorithms maintain a crucial role in finding genes in DNA. Current prokaryotic gene-finding tools have sufficiently high accuracy in predicting protein-coding genes. On average, these tools find more than 97% of experimentally verified genes [34–36]. Most genes that are not detected (false negatives) belong primarily to the "atypical" category, i.e., genes with sequence patterns that deviate from those found in the bulk of the genome [37]. However, the average accuracy of pinpointing the start positions of these genes is ~90% [36], with accuracies ranging from 81% to 98% for individual genomes.

Current approaches do not accurately represent the modes of translation-initiation found within a given genome. In particular, they often rely on simplified and heavily fine-tuned models of ribosomal binding sites, and do not account for cases of leaderless transcription (see Section 2.1.4).

In this chapter, I describe GeneMarkS-2, an *ab initio* gene-finder that relaxes some of the assumptions surrounding translation-initiation mechanisms. In particular, it can identify multiple mechanisms, e.g. RBS and leaderless transcription, within a single genome, as well as non-canonical RBS. This flexibility is advantageous, especially in genomes that frequently use multiple mechanisms for translation-initiation (e.g. *Mycobacterium tuberculosis*). I then present an exploratory analysis of translation-initiation mechanisms using GeneMarkS-2 that provides a new understanding of the diversity of these mechanisms across prokaryotic genomes.

3.1 Related Works

Comparing gene-finders must generally take into consideration the detection of both the start and stop positions of a gene. Since this work is mainly concerned with gene-start prediction, I will focus my description of related tools on their gene-start prediction mechanisms.

3.1.1 GeneMarkS

GeneMarkS [35] is a generative probabilistic graphical model based on an extension of hidden Markov models (HMMs). It learns parameters for states that represent a biologically-inspired view of a genome: genes, intergenic (non-coding) regions, RBS, gene-starts, gene-stops, etc. Given a new genome, it iteratively predicts genes and re-trains its parameters until convergence, similar to an Expectation Maximization (EM) approach. This lets it fine-tune a rough set of predictions based on properties inherent to the genome.

GeneMarkS is an *almost*-unsupervised model. The "almost" part comes from the fact that the first prediction in the iterative process is done by MetaGeneMark [38], a model which is a pre-trained on genes from a large set of genomic data. However, all remaining parts of GeneMarkS (including RBS, start/stop codon frequencies, the native protein-coding model, etc) have not been previously trained and are learned exclusively from the current genome.

With respect to gene-start prediction, GeneMarkS uses a single motif model, such as RBS, trained from the data in an unsupervised fashion. The idea is that given a set of 40 *nt* DNA fragments extracted from the regions just upstream of some gene-starts, each fragment can be thought to consist of an RBS motif (usually 4 to 6 *nt* long), and non-coding regions on either side of it. Due to the conservation of RBS motifs across sequences, we can position all the sequences so that their RBS motifs align, even without knowing what the RBS looks like. This approach was introduced in [39], and is described and extended

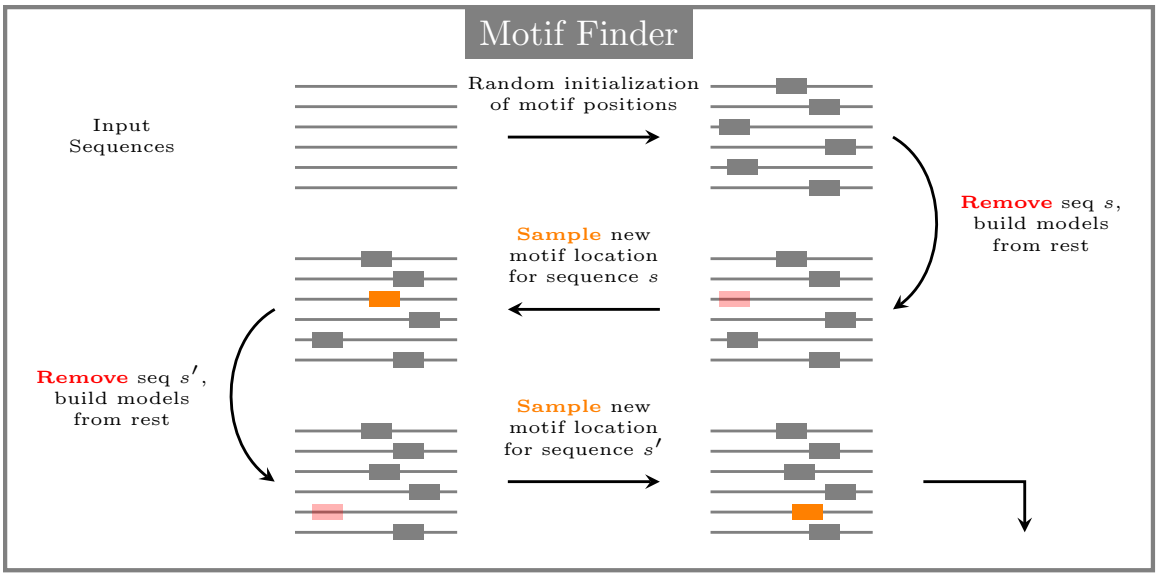


Figure 3.1: An illustration of the motif finder algorithm using Gibbs sampling. We are given 6 sequences that are upstream to gene-starts. Each sequence contains an RBS motif, but we do not know where it is. The algorithm starts by randomly assigning positions of motifs in the sequences, and then iteratively moves each motif to a new location in order to maximize some objective function. This function attempts to maximization the conservation across the final set of selected motifs.

this chapter.

At the time, GeneMarkS was shown to outperform its competitors on both gene-start and gene-stop accuracies.

3.1.2 Prodigal

In contrast to GeneMarkS, Prodigal [36] is a discriminative approach to gene prediction; it works on optimizing a hand-crafted objective function. Of particular interest here is the score for the start-codon, which has the form

$$S(n) = 4.25 * (R(n) + T(n) + 0.4 * U(n)) + C(n), \tag{3.1}$$

where $R(n)$ is the RBS score, $T(n)$ is the start-type score (e.g. ATG, GTG, or TTG), $U(n)$ is the upstream score, and $C(n)$ is the downstream score.

Prodigal's RBS model is based on pre-constructed table of possible RBS motifs (e.g.

AGGAGG, GGAG, etc). This set captures the Shine-Dalgarno (SD) RBS, i.e. the RBS with the AGGAGG consensus. To account for non-SD RBS motifs, Prodigal checks if the number of SD-RBS founds is too low; if so, it attempts to construct a table of possible non-SD motifs, and to use that as its motif scoring mechanism. As shown later, Prodigal's start model is powerful but suffers greatly when multiple types of motifs exist within a genome, especially in high-GC bacteria such as *M. tuberculosis*.

The weights for the objective function were derived from Prodigal's performance on an existing dataset of computationally labelled genes, as well as experimentally verified genes and gene-start positions for *E. coli*. The first set of genes was curated by the JGI ORNL pipeline, which is a pipeline that used Glimmer and BLAST [40] to locate missing genes that were then refined by manual expert curation.

This set was used to validate the algorithm's design decisions based on how it performed. Furthermore, another 100 genomes were added from Genbank [41] in the final tests of the algorithm. Prodigal used all these genomes to determine general rules about prokaryotic genomes, including RBS motif usages. In their tests, the authors showed that Prodigal outperformed GeneMarkS in gene-start prediction.

3.1.3 Glimmer

Glimmer [34] is another Markov-based approach to gene prediction. One of its main features is its use of interpolated Markov models (IMM). In contrast to standard Markov models, IMMs can dynamically choose the model's *order*.

A gene can be represented using a Markov model of any order. For instance, in a 5^{th} order Markov model, the next nucleotide n_{i+1} in a gene is predicted based on the distribution over the previous 5 nucleotides, i.e. $P\left(n_{i+1}|n_i, n_{i-1}, n_{i-2}, n_{i-3}, n_{i-4}\right)$. However, it is often the case that some 5-mers do not appear very frequently, and therefore their parameter estimation is not as reliable. In such cases, it is beneficial to use a lower order model (e.g. $P(n_{i+1}|n_i, n_{i-1})$) since this increases the support for parameter estimation. IMMs allow us

Figure 3.2: Principal state diagram of the generalized hidden Markov model (GHHM) of prokaryotic genomic sequence. States shown in the top panel were used to model a gene in the direct strand. Genes in the reverse strand were modeled by the identical set of states (with directions of transition reversed). The states modeling genes in direct and reverse strands were connected through the intergenic region state as well as the states of genes overlapping in opposite strands

to automatically select which order to use for any nucleotide n_{i+1}, based on the frequency of the selected base. In other words, if the base $n_{i-4:i}$ is not seen very frequently for a given setting of $n_{i-4:i}$, then a lower order model is given more weight.

Regarding gene-starts, Glimmer adopted a Gibbs-sampler approach similar to that of GeneMarkS. It uses the ELPH software [42] to generate a positional weight matrix representation of an RBS and use that to score RBS sites during prediction. In our tests on experimentally verified data, however, we still found that GeneMarkS outperformed Glimmer on 5' end predictions.

3.2 Methods

There are three components to the GeneMarkS-2 pipeline: the **models**, and the **training** and **prediction** algorithms. In this section, I provide an intuitive description of all components, highlighting the main parts related to gene-start prediction and postponing the remaining details to the appendix (see Appendix A).

The models can be thought of as a representation of a genome, detailing the parts we deem necessary (and learnable) to identify genes and gene-starts. The training algorithm is a way of learning the parameters for this representation for any given genome, and the prediction algorithm uses this trained representation to find genes in the DNA.

3.2.1 Model

If we think of our model as a generator of genomes, we can build an intuitive story using the components shown in Figure 3.2. At the highest level, the model generates a genome by iterating between intergenic (i.e. non-coding) and gene regions, with occasional regions of overlapping genes.

Within a gene, the components become more intricate. Thinking of it as a linear story, we first create a motif that will help us identify the start of a gene. GeneMarkS-2 can choose one of three types of motifs: RBS, promoters (for leaderless transcription), and an extended "upstream signature" which is a generic, last-resort model used in cases where we cannot identify an RBS or promoter for that gene. We then create the region just around the gene-start, which involves two models (upstream and downstream signatures) that capture properties around the start, as well as the start codon itself. Following that comes the entire protein-coding gene, ending with a stop codon. The result is a gene with some type of representation of its translation-initiation mechanism.

For the protein-coding region, we allow an either typical (native) or atypical representation of a gene. The latter is used to represent genes that have possibly been transferred to this genome from different species, thus exhibiting compositional properties that are different from the majority of genes in that genome. The atypical regions are represented by MetaGeneMark models (see Chapter 5).

Together, these components make up the parameters of a GeneMarkS-2 model, and the task is to "fit" these parameters to any given genomic sequence. The parameters are mostly in the form of positional and periodic Markov models, and the entire model acts as a probability distribution over all genomes.

3.2.2 Training

The training pipeline is a combination of multiple training algorithms, strung together under a parameter estimation framework akin to Viterbi Training [43].

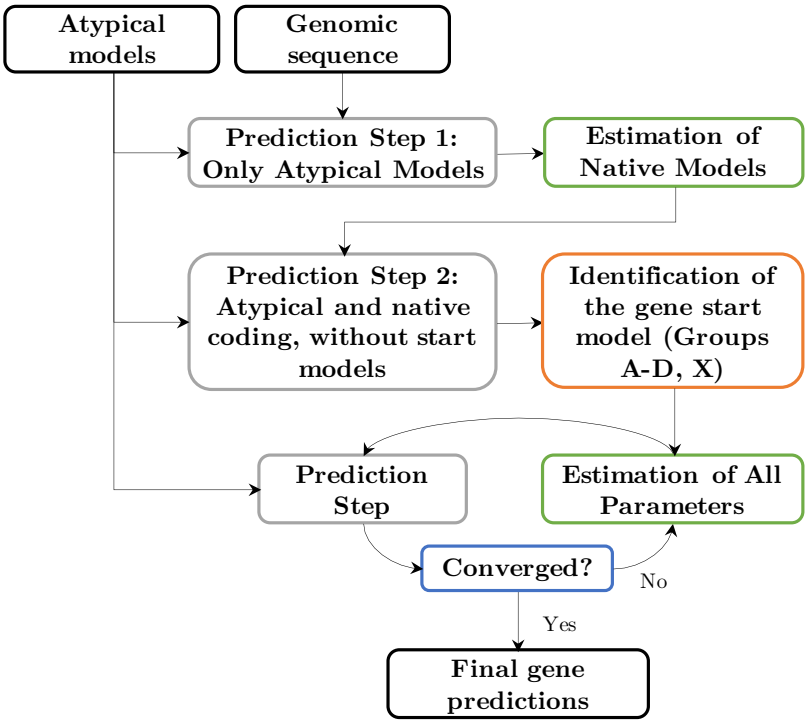


Figure 3.3: Principal workflow of the unsupervised training.

The general idea is that given a genome, we start with a crude prediction of the locations of genes (using the pre-trained, atypical MetaGeneMark models), and then alternate between parameter estimation and re-prediction of genes up until we reach convergence (or the maximum allowed number of iterations). This is shown in Figure 3.3.

I focus on the two parts related to gene-start prediction: the identification of the gene-start model, i.e. RBS, leaderless transcription, etc, and the estimation of the parameters for these models.

3.2.2.1 Determining the gene-start model (Groups A-D, X)

GeneMarkS-2 assigns a genome to one of five groups based on what type of translation-initiation mechanisms it uses. For example, some genomes almost exclusively use a ribosomal binding site for all translation-initiations, while others have a significant fraction of leaderless transcription; some genomes use non-SD RBS, and some use RBS very sparingly.

Table 3.1: Features of the regulatory site models used for genomes of Groups A-D and X. A dash indicates that a particular model was not used; -26 and -10 indicate the average nucleotide (*nt*) distance between the promoter Pribnow box and the position of the gene's start.

Groups	Features & Domain	Representative Species	RBS Consensus	Promoter Box Localization	Extended Upstream Signature
A	Leadered with SD RBS	*E. coli*	SD	-	-
B	Leadered with non-SD RBS	*B. ovatus*	non-SD	-	-
C	Leaderless & Bacteria	*M. tuberculosis*	SD	-10	-
D	Leaderless & Archaea	*H. salinarum*	SD	-26	-
X	Unclassified	*Synechocystis*	SD	-	20 *nt*

We can therefore define four groups (A through D), and a "throwaway" group X whose genomes did not pass any of the tests for the other groups. **Group A** represents the most common case, where the standard SD-RBS is used to translate almost all genes. **Group B** is again an RBS dominated genome, but where the consensus has a different structure than AGGAGG, typically rich in A's. **Groups C and D** are bacteria and archaeal genomes, respectively, where leaderless transcription is frequent; in these groups, we train both an RBS and a promoter model. Finally, **Group X** consists of the genomes that GeneMarkS-2 is not able to classify into A-D. In this case, a generic positional Markov model is used to represent their upstream region. GeneMarkS-2 still trains an SD-RBS for Group X genomes, but this is applied to a small fraction of its genes. A genome is assigned to one of these groups through a simple greedy approach described in Appendix A.3. The gist of this approach is that it attempts to build promoter and RBS models, and assigns the genome to a group based on what it could successfully build. The group configurations are summarized in Table 3.1.

3.2.2.2 Motif training

The motif models are trained separately from the remaining Markov parameters of the HMM, where the latter are estimated using a standard HMM maximum likelihood approach. Instead, the models for RBS and promoters are trained using a probabilistic sequence-alignment algorithm using Gibbs sampling.

The motif finder Gibbs3 [39] learns a probabilistic model of an *a priori* unknown motif that is present in a set of sequences. Gibbs3 was used to train the RBS model with reasonable accuracy in GeneMarkS [35].

However, the distance between motifs and their corresponding gene-starts follows a non-uniform distribution, which presumably facilitates molecular interactions involved in translation initiation. This is important because it is often necessary to discriminate between very close gene-start candidates. Gibbs3 does not consider this distribution when choosing the best motif in any given sequence.

We can account for this by explicitly including the distance between the motif and the gene-start into the objective function. I present GibbsL, an approach that penalizes motifs whose positions relative to the gene-start deviate from the norm. This "norm" is not known beforehand, and is learned by GibbsL during execution.

Formally, suppose we are given a set of N sequences $S = \left\{S^{(1)}, \ldots, S^{(N)}\right\}$, such as DNA sequences located upstream of predicted gene starts. Let $\boldsymbol{a} = \{a_1, a_2, \ldots, a_N\}$ be the vector of motif positions, where $a_n = i$ indicates the left-most position of the predicted motif in sequence $S^{(n)}$. We assume that all motifs have a fixed length W. The left and right sections of $S^{(n)}$ that does not belong to the motif are called the "background". We then find the set of positions $\boldsymbol{a}$ that maximize the probability

$$P(\boldsymbol{a}|S, \lambda) = \prod_{n=1}^{N} \left[P(a_n|S^{(n)}, \lambda)\right] \tag{3.2}$$

Intuitively, this is equivalent to finding the best probabilistic alignment of motifs across

the sequences. We can show that it is proportional to maximizing the difference between the motif and background models, and between the position distribution and a uniform distribution. Formally,

$$P(\boldsymbol{a}|S,\lambda) \propto \mathrm{KL}\left(M_{motif}|M_{bgd}\right) + \mathrm{KL}\left(M_{pos}|M_{uniform}\right) \tag{3.3}$$

where M_{motif} and M_{bgd} are the motif and background models, M_{pos} is the motif position distribution, $M_{uniform}$ is a uniform distribution over all positions, and KL is the Kullback-Leibler divergence. Proof in Appendix A.4.

Intuitively, this shows that maximizing Equation (3.2) is equivalent to finding the motif positions $\boldsymbol{a}$ that lead to the biggest (KL) difference between the motif and background models, and ensuring that the distribution of distances between the motif and the gene's 5' end is not uniform, which we know to be the case. The end result are the positions of all motifs, as well as the model representations M_{motif} and M_{pos}, which are then used in prediction. All algorithmic details are explained in detail in Appendix A.4.

3.3 Data sets

3.3.1 Sets of experimentally verified gene-starts

N-terminal protein sequencing is a standard technique to validate sites of translation initiation (protein N-terminals and gene starts). Relatively large sets of genes with validated starts were constructed for the bacteria *Synechocystis sp.* [44], *E. coli* [**Zhou2013**, 45], *M. tuberculosis* [46], and *D. deserti* [22] and the archaea *A. pernix* [47], and *H. salinarum*, *N. pharaonis* [48]. We use these sets as the main test set to quantify gene-start accuracies.

3.3.2 Set of 5,007 representative prokaryotic genomes

NCBI's prokaryotic genome collection includes 5,007 species which they describe as "representatives" of the whole database of more than 100,000 genomes [49]. These include 238

archaeal and 4,769 bacterial species to cover all genera.

Detailed descriptions of the data sets are available at http://topaz.gatech.edu/GeneMark/GMS2/.

3.4 Results

Gene-finder results for prokaryotic genomes are typically two-fold: the percentage of genes found in a genome, and the accuracy of the 5' end for those genes. To keep the focus on gene-start prediction, I defer the former to Appendix A.5 and only discuss the results pertaining to gene-start prediction. That said, it is worth noting that GeneMarkS-2 outperforms Glimmer3, Prodigal, and GeneMarkS when it comes to false-positive and false negative gene predictions; i.e., the number of false genes predicted, and the number of missed genes.

3.4.1 Gene-start accuracy on experimentally verified data

We begin by comparing gene-start predictions to the sets of experimentally verified gene-starts, which come from *A. pernix*, *D. deserti*, *E. coli*, *H. salinarum*, *M. tuberculosis*, *N. pharaonis*, and *Synechocystis sp*.

On this set, the observed error rate of GeneMarkS-2 is 4.4%, followed by Prodigal (6.1%), GeneMarkS (10.2%) and finally, Glimmer3 (13.2%) (Table 3.2). Note that while Prodigal performed better on the *E. coli* set, this set was used in the supervised training of Prodigal's gene-start prediction model [36]. This makes the comparison on *E. coli* somewhat invalid, since it doesn't capture Prodigal's generalization error on this genome.

As expected, GeneMarkS-2 made more accurate predictions for genomes with frequent leaderless transcription, in both bacteria (group C) and archaeal (group D) genomes. Interestingly, an experimental study of *D. deserti* identified 384 genes with verified translation starts, 262 of which had transcription starts annotated with differential RNA sequencing (dRNA-seq) [22]. It was experimentally shown that 167 out of the 262 genes had leaderless transcrip-

Table 3.2: Numbers of correctly predicted gene starts verified by N-terminal protein sequencing

Species	Assigned Group	# of verified gene starts	GeneMarkS	Glimmer3	Prodigal		GeneMarkS-2	
A. pernix*	A	130	125	119	**127**	**(97.7%)**	**126**	**(96.9%)**
D. deserti	C	384	315	314	334	(87.0%)	**369**	**(96.1%)**
E. coli	A	769	725	714	**751**	**(97.7%)**	740	(96.2%)
H. salinarum*	D	530	502	454	514	(97.0%)	**523**	**(98.7%)**
M. tuberculosis	C	701	572	572	620	(88.4%)	**635**	**(90.6%)**
N. pharaonis*	D	315	309	288	309	(98.1%)	**312**	**(99.0%)**
R. denitrificans	A	526	448	471	500	(95.1%)	**508**	**(96.6%)**
Synechocystis	X	96	81	79	**92**	**(95.8%)**	**92**	**(95.8%)**
(*archaea)	**Total**	3,451	3,077	3,011	3,247		**3,305**	

tion. In this genome, GeneMarkS-2 correctly predicted 34 more starts than Prodigal, which leads to a gene-start accuracy of 96% compared to Prodigal's 87%. The same is true for *M. tuberculosis*.

3.4.1.1 Comparison with GeneMarkS

Overall, the improved start prediction accuracy of GeneMarkS-2 over GeneMarkS is due to a more flexible modeling of the regulatory signals near gene starts. For instance, in *M. tuberculosis* (group C), GeneMarkS did not find a sufficiently strong RBS motif (Figure 3.6A). GeneMarkS-2, on the other hand, predicted that 40% of operons are likely to be transcribed in the leaderless fashion, with the promoter Pribnow box located at a 6-8 *nt* distance from the gene starts (Figure 3.6B). In the remaining ~60% of operons, the predicted RBS sites were also separated from the gene starts by a 6-8 *nt* distance (Figure 3.6C). Therefore, the mixture of promoter and RBS patterns located at the same distance from the gene-start did not allow GeneMarkS's motif finder to converge to an informative motif model.

Similarly, for the majority of first genes in operons in *H. salinarum* (group D), GeneMarkS-2 identified the promoter Pribnow box located at 22-24 *nt* from the gene starts (Figure 3.6E), at a distance characteristic of leaderless transcription in archaeal genomes. For the remaining FGIOs, GeneMarkS-2 identified the RBS sites at a 6-8 *nt* distance upstream of the gene starts (Figure 3.6F). GeneMarkS (as well as Prodigal and Glimmer), which assumes a sin-

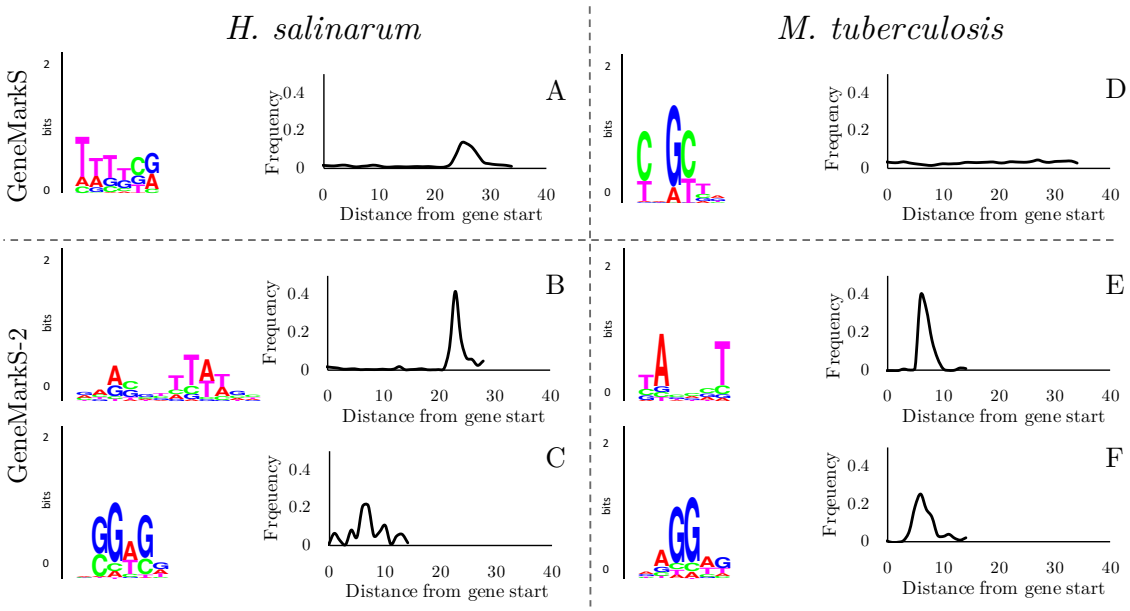


Figure 3.4: The motif models learned by GeneMarkS and GeneMarkS-2 on H. salinarum (group D) and M. tuberculosis (group C). Specifically, this shows GeneMarkS-2's ability to build multiple motif models, an RBS and promoter in genomes where leaderless transcription is frequent. GeneMarkS is able to build a weaker promoter signal in *H. salinarum*, and no signal in *M. tuberculosis*.

gle type of motif for all genes, could only derive a Pribnow box-like motif with a weaker signal and localization (Figure 3.6D).

3.4.1.2 Distribution of translation-initiation mechanisms on ~5,000 genomes

To understand the diversity of translation-initiation mechanisms in prokaryotes, I ran GeneMarkS-2 on the set of representative genomes, and mapped the group label assigned by GeneMarkS-2 to the genome's position on the taxonomy tree. The distribution of groups at the top three taxonomical levels is shown in Figure 3.5. A much more detailed tree is shown in Appendix D. Not surprisingly, the species belonging to the same clades tend to belong to the same group. What is surprising the prevalence of prokaryotic genomes using leaderless transcription and those using non-SD RBS.

GeneMarkS-2 assigns 2,935 bacteria and 39 archaea genomes to Group A (Table 3.3). Here, gene expression occurs predominantly via mRNAs with detectable SD-RBS mo-

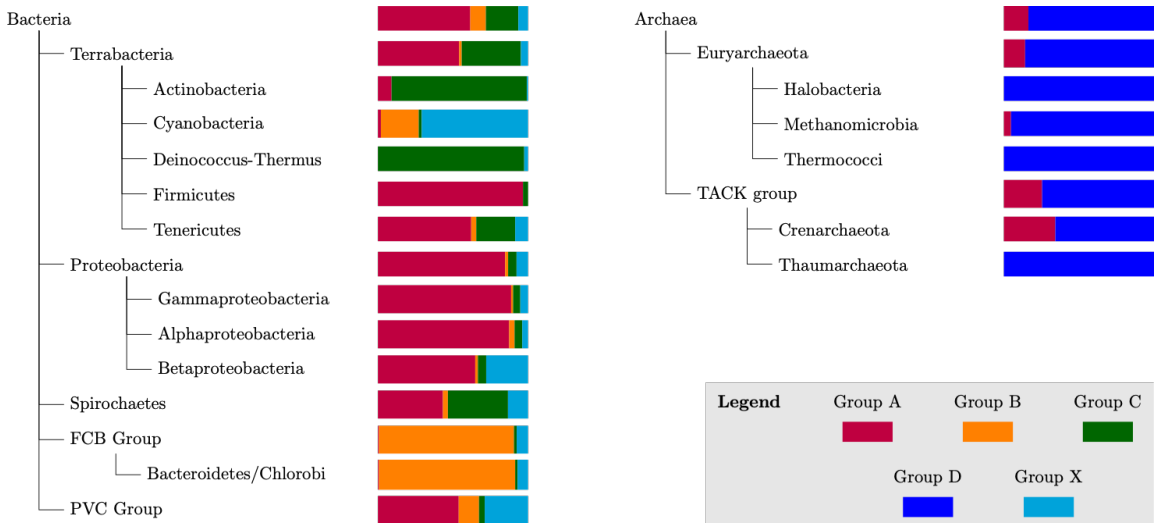


Figure 3.5: A color-coded scheme of the distribution of groups A-D and X among ~5000 representative genomes. The diagram shows the top three levels of the taxonomy trees of both archaea and bacteria.

tifs. Among the bacterial genomes in this group, 61% were Gram-negative and 39% were Gram-positive. These Gram-positive species alone make up more than half (57%) of all Gram-positive bacteria in the set of ~5000 species. Furthermore, it is only Gram-positive Actinobacteria genomes that rarely belong to group A (78 genomes out of 859, 9.1%) and mostly appear in group C; if we exclude Actinobacteria, 96% of the remaining Gram-positive bacteria belong to group A.

Next, 495 bacteria (and no archaea) genomes were assigned to Group B (Table 3.3). The characteristic feature of this category is the non-SD RBS motif. In these genomes, we see the presence of the same motif in the upstream sequences of all genes, both first and internal genes in operons, but the motif does not have the Shine-Dalgarno consensus. Since this motif is present for internal genes in operons, it cannot be a promoter because transcription does not occur near internal genes by definition. Group B species are make up most of the bacterial *FCB group* (409 out of 455, 89.9%), and are rare in Terrabacteria (1.7%) and Proteobacteria (2.0%).

Group C (1028 out of 4769 bacteria) consists of bacterial species predicted to have a frequent presence of leaderless mRNAs. These make up most of Actinobacteria (773 from

859, 90.0%) and *Deinococcus-Thermus* (37 out of 38, 97.4%) genomes, but are rare in *Proteobacteria* (104 out of 1854, 5.6%) and *Firmicutes* (36 out of 1064, 3.4%). A particularly high frequency of group C species is also seen in *Streptomycetales* (129 out of 129, 100%) and *Corynebacteriales* (197 out of 202, 97.5%), which includes *Mycobacteriaceae* (56 out of 57, 98.2%).

Similarly, group D includes archaeal species with a prevalence of leaderless mRNAs. A motif resembling the promoter box is derived for the leaderless first genes in operons, with a standard RBS motif for the remaining genes. In this experiment, 199 out of the 238 archaeal genomes are assigned to group D. In particular, some taxa have most (or all) of their members belonging to this group, such as Halobacteria (74 out of 74 species, 100%), Methanomicrobia (40 out of 42, 95%), Thermococci (21 out of 21, 100%), Thermoplasmata (11 out of 11, 100%), Archeoglobi (7 from 7, 100%), Thaumarchaeota (11 from 11, 100%) and Crenarchaeota (23 out of 35, 65%). While group D is common across archaea, note that a significant fraction of the taxon Crenarchaea, where P. aerophilum belongs, are assigned to group A. Thus, many members of Crenarchaea seem to have a low percentage of leaderless transcripts.

Finally, 311 bacterial species did not fit any of the above four groups and were assigned to group X, which is characterized by the (seeming) absence of pronounced regulatory signals upstream of most genes. Still, this absence has its own commonality. Species of this group are relatively frequent in Cyanobacteria (90 out of 127, 70.9%) and in Burkholderiales (63 out of 166, 37.9%).

The distribution of the 5,007 genomes across the groups is given in Table 3.3. We can see that the percentage of genomes with leaderless transcription and non-SD RBS is significant (32%). Figure 3.6 shows example motif models learned by GeneMarkS-2, for the four groups A-D. While previous work on individual bacterial genomes showed the existence of leaderless transcription or A-rich upstream sequences without the SD-RBS motifs, the diversity at the level shown here was, to my knowledge, not previously known

Table 3.3: The distribution of archaeal and bacterial genomes among groups A-D, and X, from the set of <5,000 representative genomes.

	Archaea		Bacteria		Total
	Number	**%**	**Number**	**%**	**Number**
Group A	39	16.4	2,935	61.5	2,974
Group B	0	0	495	10.4	495
Group C	NA	0	1,028	21.6	1,028
Group D	199	83.6	NA	0	199
Group X	0	0	311	6.5	311
Total	238	100	4,769	100	5,007

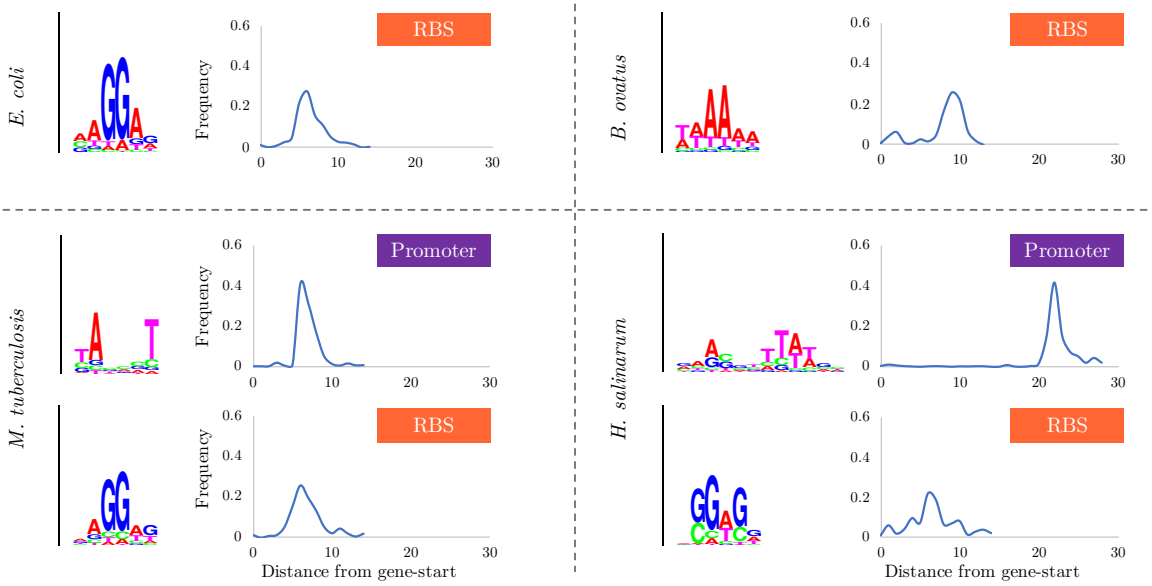


Figure 3.6: The GeneMarkS-2 motif models constructed for four sample genomes. *E. coli* (Group A), *B. ovatus* (Group B), *M. tuberculosis* (Group C), and *H. salinarum* (Group D).

or shown to this extent.

3.5 Discussion

3.5.1 Real-world leaderless transcription

Besides gene finding, GeneMarkS-2, by virtue of predicting the various types of regulatory motifs, provides a prediction of the type of transcript used, leadered or leaderless. Screening the large number of genomes led to the conclusion that leaderless transcription is a ubiquitous feature of prokaryotes (Figure 3.7).

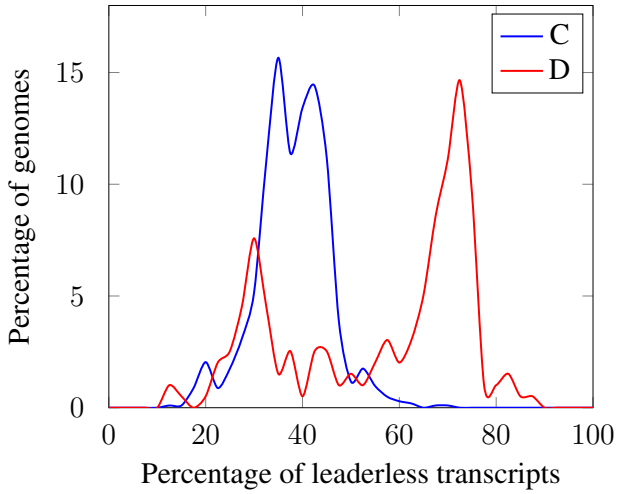


Figure 3.7: The distributions of the percentage of leaderless transcripts among all transcripts in bacterial Group C and archaeal Group D.

In many archaea, 60%–80% of operons are transcribed in a leaderless fashion. Still, in a smaller fraction of archaeal species, this percentage drops to 25%–35%, which is close to what is observed in bacteria assigned to group C (with 25%–50%).

These frequencies of leaderless transcription can be validated by comparing against those determined experimentally by dRNA-seq for several species. The dRNA-seq method identifies positions of transcription starting sites with high accuracy. Consider the dRNA-seq experiments conducted for Deinococcus deserti [22], Haloferax volcanii [6], Sulfolobus solfataricus [26], and M. tuberculosis [7]. The authors determined the lengths of upstream untranslated regions (UTRs), i.e. the sequences between the transcription start and translation start sites, The cases where this length is short (e.g. $<6nt$) indicate the presence of leaderless transcription.

For simplicity, consider the subsets of genes where the positions of the annotated gene starts match positions predicted by GeneMarkS-2. In *D. deserti*, out of 1707 such genes, ~62% were predicted as leaderless by GeneMarkS-2 and 62% were observed leaderless by dRNA-seq, i.e., the UTR length was <6 *nt*. In *M. tuberculosis*, these percentages are 42% and 34% out of 1310. Similarly in archaea, we have 86% and 82% out of 1406 genes in *H. volcanii*, 78% and 76% out of 859 in *S. solfataricus*. This shows that GeneMarkS-2's

predictions are similar to those from dRNA-seq experiments.

The species with a large proportion of leaderless transcripts were all classified as group C [7, 21, 22, 24] or group D [6, 17, 18, 25, 26]. Note, however, that for species where the dRNA-seq experiments found small numbers of leaderless transcripts [8–12, 14, 15, 21], GeneMarkS-2 classifies them as group A.

Experiments with *Synechocystis sp.* demonstrated the prevalence of leadered transcription [50]. However, GeneMarkS-2 detected an SD-RBS motif in fewer than 15.5% of its genes. Other experiments have shown that mutating A-rich sequences situated 15–45 *nt* led to changes in gene expression [51]. Nonetheless, the translation-initiation mechanisms for the majority of *Synechocystis sp.* genes remains unknown.

In *E. coli*, all three types of translation-initiation (SD RBS, non-SD RBS, and leaderless) are present [52–54]. Similar observations were made for other species, and it was shown that the distribution of the number of genes controlled by each of the mechanisms could vary significantly [55]. For the types rarely present in a given species, GeneMarkS-2 is not able to build these models due to the insufficient size of the training set.

3.5.2 Regulatory Motifs in group B

GeneMarkS-2 currently assigns 495 out of 4769 bacteria (and none out of 238 archaea) to group B. Consider *Bacteroides ovatus* as an example. Its 16S rRNA features the standard "anti-SD" pattern; that said, SD-matching sequences appear upstream of some gene starts (fewer than 3% of genes). GeneMarkS-2 identifies the A-rich, non-SD motif localized at ~9 *nt* from the gene-start (Figure 3.8). The A-rich sequences appear in the upstream regions of the majority of *B. ovatus* genes. Since mutating the A-rich regions reduced gene expression levels, it was proposed that these regions are an important part of the translation-initiation mechanism [56].

GeneMarkS-2 assigns 90% of *Bacteroidetes/Chlorobi* species to group B (408 of 450). While not much is known about the non-SD translation mechanisms in bacteria, the cluster-

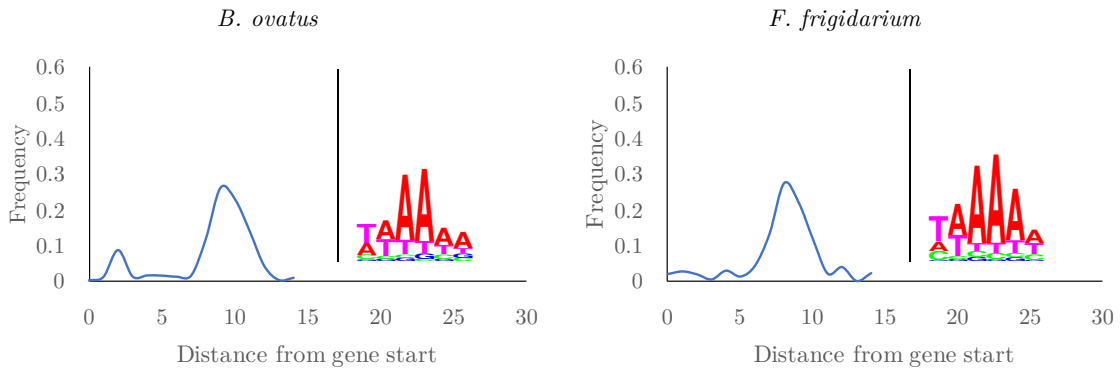


Figure 3.8: The motif logos and spacer length distributions of *Bacteroides ovatus* and *Flavobacterium frigidarium* (group B).

ing of the species assigned to group B within particular clades lends additional credibility to the results (Figure 3.5).

Particularly in *Bacteroides*, which includes *B. ovatus*, 21 out of the 23 species were assigned to group B. In these genomes, GeneMarkS-2 found motifs similar in sequence and localization patterns to those of *B. ovatus*.

Similarly, 30 out of the 30 *Flavobacterium* species (a genus from *Bacteroidetes/Chlorobi*) are assigned to group B. They possess 6-*nt*-long motifs similar in sequence and localization patterns to those in *B. ovatus* (Figure 3.8).

The genomes from these genera are closely related, but there are still some differences in the derived motifs. In particular, *Bacteroides* motifs tend to have a few strongly conserved A nucleotides close to the translation start, hence the secondary peak in the spacer length distribution at a ~3 *nt* distance from the start (Figure 3.8). In *Flavobacterium*, the "core" motif with consensus TAAAAA is more pronounced than in *Bacteroides*. However, *Flavobacterium* is missing the strongly conserved A nucleotides near the gene start. Therefore, its "core" motif is situated at the end of the 15 *nt* window rather than at the 3 *nt* position as in *Bacteroides* (Figure 3.8). The shift in the position of the "core" motif leads to a change in the spacer length distribution, which now has a peak at ~7 *nt*. The consistency of this observation was tested for all 21 *Bacteroides* and 30 *Flavobacteria*. In addition,

the 6nt "core" motif was not easy to detect in *Prevotella* (a close relative of *Bacteroides*) when the motif width was set to 6 *nt*. However, setting the motif width to 15 *nt* led to a similar motif. Notably, unlike *B. ovatus*, other group B species may have a 16S rRNA with a mutated or truncated tail [57].

In a recent publication [3], species with leaderless transcription and non-SD translation initiation were assigned to the same class. GeneMarkS-2 is able to make a distinction between these cases.

3.6 Conclusion

The accuracy measurements of gene and gene-start prediction is still a debated topic. This is due in part to the small number of experimentally verified gene-starts. That said, GeneMarkS-2 provides a marked improvement in gene-start accuracy, as well as the ability to characterize a genome by its translation-initiation mechanisms. This improvement is clear when considering all genomes of groups C and D in the verified set, showing that the dual model provides more accurate gene-start predictions.

CHAPTER 4
BOOSTING GENERALIZATION USING INDEPENDENT SOURCES OF EVIDENCE

A major obstacle hindering progress in gene-start prediction is the lack of a large ground-truth dataset; i.e. a set of experimentally verified gene-starts. Without it, we cannot easily learn features that characterize regions around gene-starts. It also causes uncertainty during model design; for example, we saw in Chapter 3 how previous algorithms ignored factors that influence translation-initiation, such as leaderless transcription and non-SD RBS. Most importantly, without a ground-truth set, comparing gene-start prediction algorithms becomes a difficult challenge.

In this chapter, I will discuss how using independent sources of evidence can provide additional support for gene-start predictions beyond the existing ground-truth dataset, which in turn helps mitigate the problem caused by the lack of verified gene-starts.

4.1 Motivation

In Chapter 3, GeneMarkS-2's gene-start accuracy was measured by comparing its predictions to a set of experimentally verified gene-starts. In addition, I provided a more qualitative analysis of how GeneMarkS-2's gene-start model generalizes by visualizing the motifs it learns (e.g. RBS, promoters), showing that the patterns cluster together on the taxonomy tree in a way that agrees with an evolutionary point of view.

Despite all this, the following questions remain largely unanswerable:

1. How well does GeneMarkS-2 perform on gene-start prediction?

2. Given a set of gene-start prediction algorithms, which performs best?

The answers come down to two important concepts in predictive modeling, overfitting and

generalization. Consider the set of experimentally verified gene-starts. Overfitting is when an algorithm is optimized to perform well on this ground-truth dataset, but does not perform well on data it has not seen before. Generalization is the ability of the model to perform as well on unseen data as it does for data it has seen.

In our case, the set of experimentally verified gene-starts is small and not representative of all genes. It currently holds just over 3,000 genes coming from 7 genomes (3 archaea, 4 bacteria) that exist in the mid- to high-GC range. Comparatively, *E. coli* alone has more than 4,000 genes, and the current number of annotated genes in RefSeq is more than 700 million, coming from 180,000+ genomes.

4.1.1 Proof in numbers

We can demonstrate that the available set of verified gene-starts is not a good measure of gene-start accuracy by comparing the relative behavior of the algorithms on this set to a larger set of genomes. When tested on the verified set, which consists of a handful of genomes, gene-start prediction algorithms achieve an average accuracy of 95% per genome [36, 58]; in particular, these tools disagree with each other by 6% on average per genome.

However, this disagreement is much larger if we consider the 5,488 representative genomes selected by NCBI. On this set, GeneMarkS-2 [58], Prodigal [36], and PGAP's annotation [59] can disagree by 15-25% of gene-starts per genome (Figure 4.1).

The reasons for these differences may be due, in part, to simplified models of the genes' upstream regions. For example, Prodigal uses a single-motif model that is unable to capture both promoter for leaderless transcription and RBS in a single genome. On the other hand, GeneMarkS-2 was built to handle both leaderless transcription and RBS within a single genome. Naturally, this allows GeneMarkS-2 to more easily detect gene-starts with leaderless transcription. In Chapter 3, we saw that GeneMarkS-2 detects a significant number of leaderless transcripts in more than 83% of archaea and 21% of bacteria. In bacteria, this was common in high-GC genomes such as in *Actinobacteria*, and those exhibit large

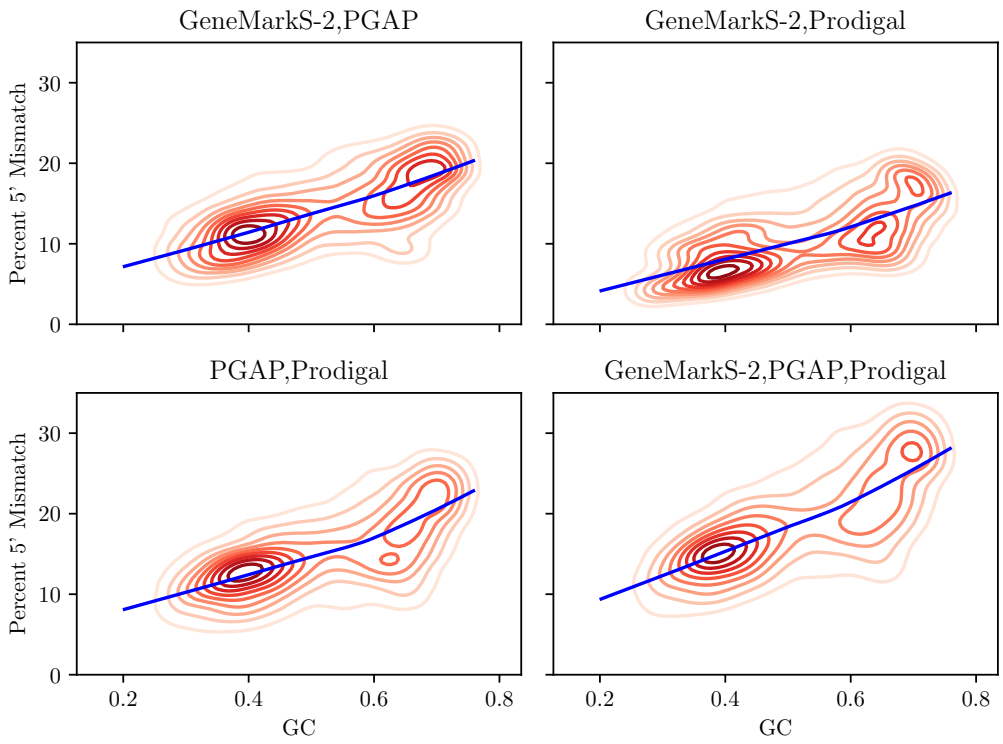


Figure 4.1: Disagreements of three tools (Prodigal, GeneMarkS-2, and NCBI's PGAP) in gene start predictions. The analysis is done over an NCBI collection of 5,488 representative genomes. In each graph, the percentage of mismatching 5' ends (per genome) is computed by taking the number of genes where *at least one* of the tools has a mismatching 5' end to the other(s), divided by the number of genes that are predicted by all tools for that graph. Each shows the distribution of genomes (contours) and the average of the percentages as a function of GC content (solid line).

percentages of disagreement (Figure 4.1).

The difference between PGAP and the other tools is more difficult to understand, since PGAP frequently uses a comparative approach while GeneMarkS-2 and Prodigal are *ab initio*. Nevertheless, the disagreement is clear, especially in high-GC, which means that the accuracies measured on the set of verified gene-starts may not translate to the large majority of genomes.

4.2 What do we do when we don't know what's right?

The most obvious solution to this problem is to *get more verified data*. Unfortunately, since the 1990s, the demand for N-terminal sequencing experiments decreased significantly in favor of mass spectrometry. While the latter could be performed much more effectively, it often lacked the precision of determining the correct start of a protein compared to N-terminal sequencing [27].

In light of this, we are now left largely in the dark. In this section, I will discuss our next best alternative to ground-truth data: supporting predictions using independent sources of evidence.

4.2.1 Anecdotal case for independence: The age of the Earth.

The concept of independent experiments and analysis is not new, and has in fact been used to weed out, correct, and refine scientific hypotheses for hundreds of years.

In the early 20th century, Arthur Holmes published "The Age of the Earth," where he provided an estimate for the earth's age using radiometric dating: 1.6 billion years old. He later revised his estimate to 4.5 billion years, which is close to what we consider to be true today.

Since then, the age of the earth has been independently verified by analyzing rocks from different continents on earth. Scientists then resorted to different types of data and methodology by using mass spectrometry to date crystals that formed in volcanic eruptions. Not satisfied with plunging into the depths of volcanoes, scientists looked outwards beyond our atmosphere. Using uranium-lead techniques and the understanding of the relationship between Earth and the other inhabitants of its solar system, Earth was dated to be 4.54 billion years old. More recently, genetic studies have placed the age of our last universal common ancestor (LUCA) to roughly between 3.5 to 4.5 billion years old. This estimate was fortunate for biologists (not to mention for LUCA).

Absent any blueprint detailing the true age of the earth, our best approach was to corroborate predictions by independent analysis. The more independent the experiments (with their use of different data, different techniques, and different assumptions about the world), the more weight the final prediction held. If, at any point, two pieces of evidence conflicted, then one (or both) would eventually need to be laid to rest.

4.2.2 Mathematical case for independence

While there's nothing like a good story, there is fortunately a more formal approach to show that independent predictions result in more trustworthy results. I describe it here in the context of gene-start prediction (albeit in a simplified environment).

Consider the task of selecting the true start of a gene from a list of candidates. Two algorithms, A_1 and A_2, each makes a single prediction, x_1 and x_2, respectively, with gene-start accuracy rates of $\mathrm{Acc}(A_1)$, $\mathrm{Acc}(A_2)$. Here, accuracy is measured as the number of correct predictions, divided by the total number of predictions.

Consider a single gene with candidate starts $C = \{c_1, \ldots, c_{|C|}\}$ and a **single true start** $s \in C$. Our goal is to answer the following question:

> Given that algorithms A_1 and A_2 predict the same start y (i.e. $x_1 = x_2 = y$), what is the probability that y is the true start (i.e. $y = s$)? Formally

$$p(y = s | x_1 = y, x_2 = y) = ? \tag{4.1}$$

We first define the accuracy and error rates in terms of probability:

$$p(x_i = y | y = s) = \mathrm{Acc}(A_i)$$
$$p(x_i = y | y \neq s) = 1 - \mathrm{Acc}(A_i) = \mathrm{Err}(A_i)$$

Using the above formulation, we can solve for Equation (4.1) for three different cases,

where A_1 and A_2 are:

1. random algorithms (i.e. select a start uniformly at random)

2. completely independent

3. completely dependent (e.g., the same algorithm)

Deriving (Equation 4.1) for randomness (P_R), independence (P_I), and full dependence (P_D) conditions (see Appendix B.1 for proof), we get

$$P_I = \frac{1}{1 + (|C| - 1)\frac{\mathrm{Err}(A_2)\,\mathrm{Err}(A_1)}{\mathrm{Acc}(A_2)\,\mathrm{Acc}(A_1)}} \tag{4.2}$$

$$P_D = \frac{1}{1 + (|C| - 1)\frac{\mathrm{Err}(A_2)}{\mathrm{Acc}(A_2)}} \tag{4.3}$$

$$P_R = \frac{1}{|C|} \tag{4.4}$$

Using the fact that $p(y \neq s|x_1 = y, x_2 = y) = 1 - p(y = s|x_1 = y, x_2 = y)$, Figure 4.2 shows the increase in probability of the mutual prediction being false as the number of options to select from increases. More importantly, it shows the effect of adding a second, independent prediction on the overall error rate, indicating that the largest improvement can be gained by maximizing independence between the algorithms.

Note that the real error rates of gene-start prediction are more difficult to model theoretically, and the above makes simplifying assumptions (e.g., on the uniformity of selecting any of the incorrect starts when a wrong prediction is made). However, the central point, that achieving maximum independence between the algorithms improves their joint performance, still holds.

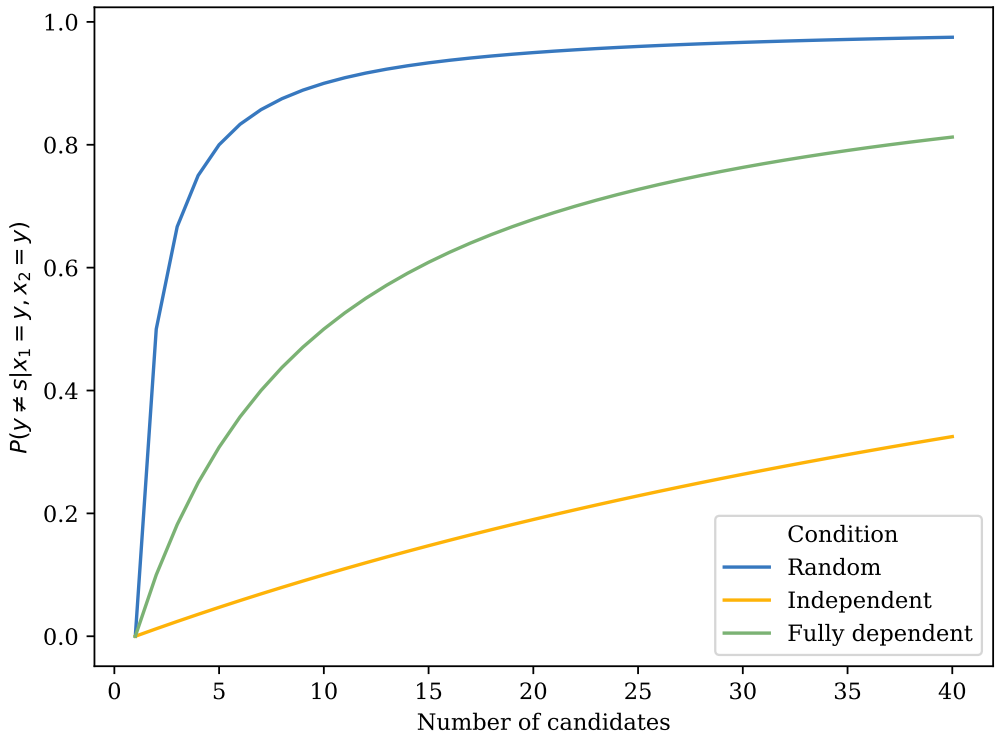


Figure 4.2: The probability that a selection made by both algorithms A_1 and A_2 is incorrect, as a function of the number of candidates to choose from. The plots are shown for the three dependency conditions between A_1 and A_2.

4.2.3 The levels of independence

It is worth taking a moment to highlight where independence should be considered or, put more sinisterly, where dependence can sneak in.

Gathering multiple independent predictions amounts to more than simply using different algorithms. We have to control for variables such as data bias (e.g., learning from the same dataset), model assumptions (e.g., assuming linearity or similar Markov properties in both algorithms), and the limits of human domain knowledge (e.g., deciding which features to include, and how to represent them).

Consider again the task of predicting the start of a gene in genomic sequence G using two different algorithms, A_1 and A_2. A common approach, such as in GeneMarkS-2 and Prodigal, is to learn model parameters from G and use them to predict the gene-start. How-

ever, this limits the models by the information, noise, and biases that exist within genome G, increasing their risk of suffering from the **same type** of **data bias**. For example, if most genes in G have a ribosomal binding site (RBS) upstream of the gene-start, then the models are likely to search for a candidate gene-start that has an RBS-like sequence upstream of it. Therefore, while both A_1 and A_2 might perform well on most genes, they might fail (due to the same data bias) on the few genes in G that do not have an RBS.

Moreover, current tools rely to some degree on the human understanding of the RBS and gene-start related features. Fine-tuning of these features may have been done on a similar set of experimentally verified gene-starts, or another set of well-studied genomes (such as *E. coli*). However, as is shown in [58], our assumptions regarding such sites have been rather simplistic in the past, and are still likely to be incomplete or biased, given the limited set of verified data available to us. As such, the same **human-induced biases** applied to these tools can result in similar biases in their final predictions.

One way to overcome some of these issues is to design an approach from a completely different starting point. For example, instead of learning features from genome G and then predicting the start of gene g, we can use relative (orthologous) genes of g that come from other genomes. This **similarity-based** approach relies on direct comparisons between a gene and its orthologs coming from other species, rather than building statistical properties from genes within a single genome. It has the advantage of using a separate and somewhat independent type of data.

Moreover, the type of biological knowledge used in both tools is also different. For instance, GeneMarkS-2 builds species-specific models of translation-initiation factors, coding and non-coding regions, etc. while a comparative approach typically relies on conservation patterns and evolutionary distances between orthologs. This removes the genome-based biases that come from training on a single genome and instead relies on information regarding protein conservation across multiple species. Figure 4.3 shows the difference in the scope of information used by GeneMarkS-2 and a similarity-based approach, where the

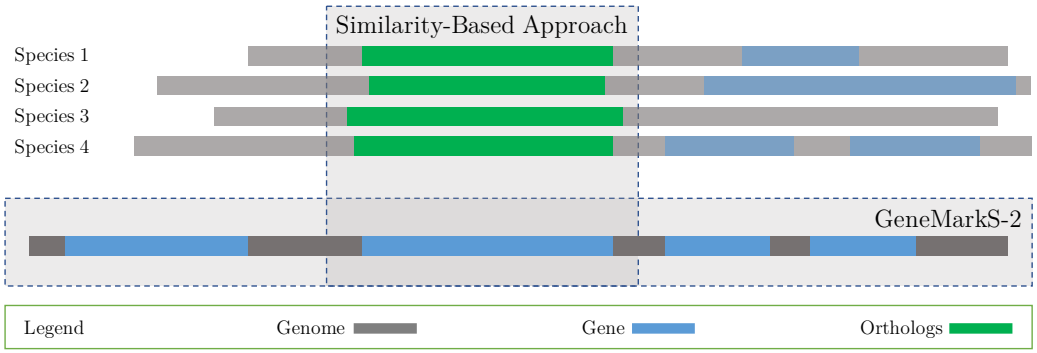


Figure 4.3: This figures shows the difference in data information that a similarity-based approach would use compared to GeneMarkS-2. The information needed by the similarity-based approach (i.e., orthologs of a gene from different species) is different from that which is needed by GeneMarkS-2 (i.e., the genome of the species under consideration).

only joint data is the gene itself that will be used for prediction.

Current similarity-based approaches often rely on protein databases, aligning the full length of these proteins to detect protein conservation. Since these databases are often partially constructed by tools such as GeneMarkS-2, used in NCBI's PGAP, and Prodigal, used by the Joint Genome Institute (JGI), this runs the risk of propagating errors and biases into new predictions. Furthermore, many similarity-based approaches rely on scores computed by other tools. For example, in [60], they use protein alignment with scores computed by Prodigal to better direct their selection, but this biases new predictions towards Prodigal's.

4.2.4 Combining independent gene-start predictions

In this chapter, I present **StartLink**, a purely similarity-based gene-start prediction algorithm designed with **independence** in mind. Specifically, (i) it does not use existing gene-start predictors as part of the algorithm, (ii) it eliminates existing gene-start biases by "extending" annotated genes into their upstream region up to the furthest valid candidate gene-start, thus removing all information of the existing gene-start selection, and (iii) it does not use translation-initiation models, of which our knowledge is not complete and are used in many *ab initio* gene-finders.

I then show that the set of gene-starts that StartLink and GeneMarkS-2 agree on has

an error rate of 1%. This set is called StartLink+. I show that StartLink+ can serve as a reference set for improving prokaryotic genome annotation. Furthermore, StartLink can serve as a standalone predictor in cases where GeneMarkS-2 and other *ab initio* tools do not perform well due to insufficient training data, such as in short contigs from metagenomic reads.

4.3 Related Works

4.4 Methods

This section is divided into three parts. First, I define the metrics that will be used to measure algorithm performance. Second, I discuss how two independent algorithms can be combined for gene start prediction: a new similarity-based start-prediction algorithm (StartLink) and a previously developed *ab initio* gene-finder (GeneMarkS-2). Finally, I present the details of StartLink's design.

4.4.1 Metrics for gene-start performance

There are two types of metrics that we are concerned with. The first tracks the fraction of genes for which a gene-start prediction is made, and the second deals with the correctness of these predictions.

Given a set of predicted genes P and a base (usually ground-truth) set B on which to compare, let **coverage** (Cov) be the number of genes in P that exist in B, irrespective of whether their gene-starts match. On the other hand, the **accuracy** (Acc) and **error** (Err) rates are the fractions of gene-starts in P that were correctly and erroneously predicted,

respectively. Formally,

$$\mathrm{Acc}(P,B) = 100 * \frac{M_s(P,B)}{M_g(P,B)} \tag{4.5}$$

$$\mathrm{Err}(P,B) = 100 - \mathrm{Acc} \tag{4.6}$$

$$\mathrm{Cov}(P,B) = 100 * \frac{M_s(P,B)}{|B|} \tag{4.7}$$

where $M_s(P,B)$ and $M_g(P,B)$ are the number of gene in P that match B by gene-start and gene-stops, respectively. As a reminder, a match by gene-stop means that the gene is found, irrespective of whether its start is predicted correctly.

4.4.2 StartLink+: Combining StartLink and GeneMarkS-2

Let StartLink+ be a "tool" that selects genes where StartLink's and GeneMarkS-2's gene-start predictions match. This may seem simplistic at first, but the underlying characteristics of informed and independent predictions provides support for this approach, as shown in Section 4.2.2.

The idea behind StartLink+ is that instead of trying to improve gene-start accuracy, which is difficult because we have no satisfactory way to measure it, we remove predictions that are more likely to be false. This reduces the total number of predictions, but those that remain have a high accuracy. The goal in designing StartLink+ is to maintain as high a coverage rate as possible, without sacrificing gene-start accuracy.

4.4.3 StartLink

Consider the task of predicting the start of a single gene. Given a query gene's previously-annotated start and stop, our goal is to re-position its start. The algorithm, shown in Figure 4.4, can be broken down into three steps.

1. Gathering a set of target genes (e.g., orthologs).

2. Filtering this set and setting up the multiple sequence alignment (MSA).

3. Running the gene-start search algorithm on the MSA.

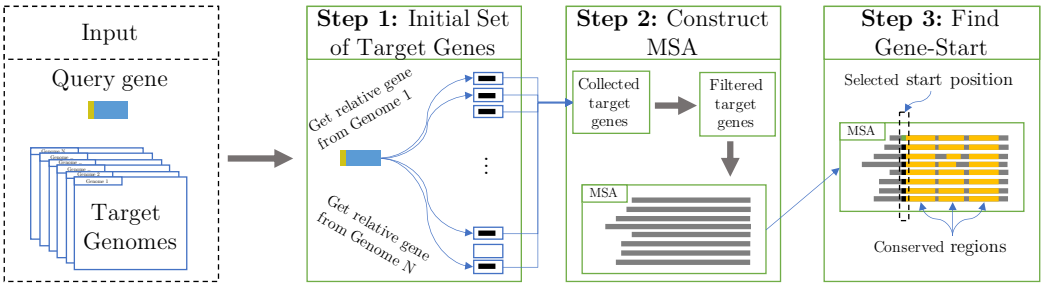


Figure 4.4: A high-level schematic of the StartLink pipeline, showing the steps of gathering orthologous genes, building a multiple sequence alignment (MSA), and using it to search for the true gene-start.

4.4.3.1 Step 1: Finding remote orthologs

Given a query gene and a database of target proteins, described in more detail further down,, StartLink uses DIAMOND's BLASTp [61] to find a set of genes that have a significant similarity to the query. The goal is to gather a set of related sequences that, after further filtering, will exhibit enough similarity around the gene-start region to allow for start detection.

We impose minimal constraints at this stage to ensure that we do not accidentally filter out useful sequences before analyzing them further. During this search, a target sequence is removed if the alignment does not cover 80% of both the query and target sequences. This restriction helps eliminate cases where, for instance, the shorter query protein aligns to a domain of a target protein not close to the gene-start, or vice versa.

4.4.3.2 Step 2: Filtering and constructing the MSA

Target sequences selected for a given query are then processed by a series of filtering steps to build a stable multiple sequence alignment (MSA) from which the gene-start will be

inferred. The filters consider query-to-target and target-to-target evolutionary distances, and whether any given target introduces a significant number of gaps to the MSA.

Evolutionary distances between each target and its corresponding query are computed using the Kimura 2-parameter model [62], which is defined as

$$d_{AB} = -\frac{1}{2}\ln\left[(1-2P-Q)\sqrt{1-2Q}\right] \tag{4.8}$$

where P and Q are the fractions of positions in the sequence that differ by a transition and transversion mutation, respectively. Targets that fall outside the range [0.1,0.5] are removed. These thresholds were selected based on the algorithm's performance and by a manual and independent analysis of resulting MSAs from randomly selected genomes.

Kimura distances are usually computed using a global alignment of two genes. It turns out that for closely related relatives, a local alignment, which is already given by the BLASTp output, provides a reasonable proxy for our purpose, relieving us from the expensive task of realigning sequences (see Appendix B.5).

Given this reduced set of sequences, we now attempt to construct an MSA suitable for gene-start inference. In doing so, we look for the following properties:

1. The existing gene-start annotations in the database do not bias the MSA. This step is crucial since we want to remove any bias and errors that come from existing tools.

2. Target sequences should not be very similar to each other since two exact datapoints bias our dataset without providing more discriminative power.

3. There is as much stability near the true gene-start region as possible. Generally, a large number of gaps in the alignment just downstream of the gene-start indicates that the alignment, in that region, may not be as easy or reliable to analyze.

First, StartLink extracts the longest open-reading-frame (LORF) of each gene and translate it into its amino acid sequence. The translated LORFs, and **not** the database's annotated

genes-starts, are then used in the MSA, thus eliminating gene-start biases coming from the database.

Second, 50 sequences are randomly selected from the set of translated target LORFs. Along with the query, these are used to construct an MSA using Clustal Omega [63]. The MSA is then used to infer pairwise alignments and compute Kimura distances between targets. For any pair of sequences with a distance that doesn't fall into the accepted Kimura distance range, one of the sequences is removed. Finally, any target sequence that creates a large number of consecutive gaps in the first half of the MSA is removed (see Appendix B.5 for details).

Note that every time sequences are filtered out, the MSA is regenerated using the remaining sequences and the filtering steps are applied again. The number of sequences in the final MSA is frequently between ~10 and 50, with the average varying per clade (Figure 4.18). For those with lower numbers, e.g. close to 10, the MSAs still contain informative features for accurate gene-start prediction (Section 4.6.6).

4.4.3.3 Step 3: Algorithm for Gene-Start Detection

In this step, we determine the most likely position for the query's gene-start using features of the MSA. Basically, we use identity scores in order to identify conserved regions in the MSA, which can help discriminate between coding and non-coding regions.

This search is broken down into three consecutive steps. We first try to determine if the true start is at the LORF by checking if only one possible candidate exists upstream of the first conserved block in the MSA. This step is very accurate because identifying coding regions using block conservation, given our previous sequence filtering, turns out to produce low false positives. If that fails, we then search for a start near the end of the upstream gene; this is inspired by studies showing preference for gene-starts near the ends of the upstream genes, which increase translation efficiency. Finally, if a start is not selected in either of the previous steps, we run a general search based solely on the conservation of

5' ends across candidates. These three steps are ordered by reliability, prior information regarding upstream genes, and finally a more general metric for the remaining cases.

Step A: Detection of conserved blocks with a single upstream 5' end Given an MSA constructed from the translated LORFs of a query and its targets, the algorithm searches for the left-most block of conserved amino acids, under the assumption that this sequence alignment block will consist of protein-coding regions. We also require this block to not have any overlap with the upstream gene in cases where regions of the upstream gene appear in the MSA. If a single candidate gene-start in the query exists upstream of the block, then it is predicted as the query's gene-start. If not, the algorithm proceeds to step B. Note that this step does not require that the gene-start is conserved across orthologs. Instead, it relies only on the joint condition that a block is conserved, indicating a coding region, and that a single start candidate appears upstream of it in the query.

To detect a protein-coding block of length r (where $r = 10aa$ not including), we define an identity score as the fraction of amino acid matches for all pairwise sequences in that region. The score is computed over the r positions in the MSA that do not have gaps in the query sequence. The score of a block is given by

$$S_{blk}(i, r) = \frac{1}{r \times (N-1)^2} \sum_{m \neq n} \sum_{j \in J(i,r)} H(m, n, j) \tag{4.9}$$

where $J(i)$ is the set of r positions downstream of position i that do not have a gap in the query, $H(m, n, j)$ is 1 if and only if sequences m and n match each other at position j in the alignment, and N is the total number of sequences in the alignment. If $S_{blk}(i, r)$ is above 0.5, we label the block as a conserved protein-coding region. The 0.5 threshold is selected to get an uninformed, majority-vote approach, which is a reasonable option in cases where little ground-truth data is available for fine-tuning.

Step B: Detection of gene starts in the presence of overlapping genes If a LORF overlaps with the end of its upstream gene, then such an overlap is likely to appear in orthologs

at a sufficiently close evolutionary distance (Figures 4.10 and 4.11). Moreover, [64] suggest that a ATG, GTG, or TTG codon located near the end of the upstream gene has a higher likelihood of being the true start, since the ribosome can efficiently reassemble at the translation start site of the downstream gene upon completing the translation of the upstream gene. Therefore, if the end of the upstream gene overlaps with the query's LORF, the algorithm searches for a nearby gene-start candidate that is conserved in the MSA.

Specifically, the search is conducted in the query sequence within a 9 *nt* radius around the upstream gene-stop. For a given position in the MSA, the conservation score is defined as the fraction of targets that have a gene-start candidate within a 2 *aa* radius around that position.

Mathematically, the start end identity score for position i in a neighborhood of radius x is defined as

$$S_{5'}(i,x) = \frac{1}{N}\sum_{j=1}^{N}\left(G(i,j,x) - P(i,j)\right) \tag{4.10}$$

where

$$G(i,j,x) = I\left\{\left|\{\text{ATG,GTG,TTG}\}\bigcap \text{Neigh}(i,j,x)\right| \geq 1\right\}$$

Here, $I\{\cdot\}$ is the indicator function, $|\cdot|$ computes the size of a set, and $\text{Neigh}(i,j,x)$ is the set of codons in a neighborhood of x amino acids around position i in sequence j; i.e. if x is 1, this returns the codons representing the amino acids at positions $i-1$, i, and $i+1$ in sequence j. In other words, $G(i,j,x)$ is 1 if an ATG, GTG, or TTG exists in the region, and 0 otherwise.

Finally, $P(i,j)$ penalizes the the appearance of start codons synonymous to ATG, GTG, or TTG, but not able to serve as start codons. In particular, $P(i,j) = 1$ if such a codon exists in position i of sequence j in the MSA, and 0 otherwise. These synonymous codons will tend to appear within the coding region. If $S_{5'}$ is larger than 0.5, we move on to Step C-2. Otherwise, we move to the next candidate. If all candidates have been exhausted, then the algorithm quits without selecting any candidate as start.

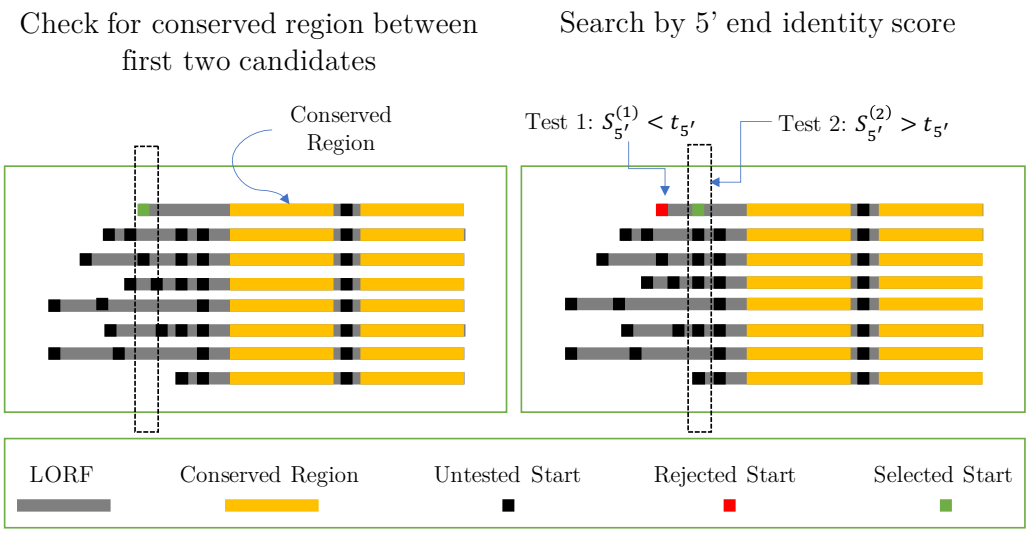


Figure 4.5: The use of multiple sequence alignment (MSA) to identify a start of a gene in the query sequence (top sequence in each MSA). This expands on step 3 in Figure 4.4. **Left panel:** Step A: the left-most conserved block is detected, and a single start candidate is located upstream of it. **Right panel:** Step C: Candidate start codons are screened to find those with conservation score above $t_{5'} = 0.5$ (see Appendix B.3)

Step C: Detection using gene-start identity score This part constitutes two sub-steps.

Sub-step C-1: When multiple gene-start candidates exist upstream of the left-most conserved amino acid block, the algorithm screens candidates in the order they appear, from the LORF downstream, by checking their gene-start candidate score, $S_{5'}$. If $S_{5'}$ is larger than 0.5, it proceeds to Step C-2. Otherwise, it moves to the next candidate. If all candidates have been exhausted, the algorithm quits without selecting any candidate as start.

Sub-step C-2: In sub-step C-1, we examine candidates in order of their position starting from the LORF. Therefore, wrong predictions at this step will likely be biased to those upstream of the true gene-start. To mitigate this effect, the algorithm examines a 10 *aa* region downstream of it for a candidate with a higher $S_{5'}$ score. If one exists, the algorithm checks if the block region (of any length between 1 and 10 *aa*) between the two candidates is conserved. If it is, the more upstream candidate is selected; otherwise, the downstream candidate is selected as the start (see Figure 4.5).

4.4.4 StartLink+

Since the probability of an event that two sufficiently accurate independent tools make exactly the same wrong prediction is proportional to a product of probabilities of an error of each tool, we run both GeneMarkS-2 and StartLink on a given genome and select a set of genes for which both tools give the same predictions. We call this set of gene starts an output of StartLink+. The disadvantage of this approach is that by far not all the genes received gene start predictions, however, as we show in the results section for a significant number of genes this combination of tools generates highly accurate gene start predictions.

4.5 Datasets

From an algorithmic perspective, the data is split into queries and targets. Query genes and genomes are those for which we want to make a gene-start prediction, and targets are used in making these predictions (e.g., orthologs of query genes).

4.5.1 Target Databases

As of November 4, 2019, NCBI's RefSeq database had over 183,689 annotated genomes. Using all genomes as our target database would cause an unnecessary computational overhead. Instead, we limit our target set only to those existing under some ancestral clade relative to the query.

For example, if we take *Escherichia coli* as the query genome, we can select *Enterobacterales* as the ancestor clade. Then, all translated LORFs of annotated genes from the genomes of the species in the clade are extracted and a protein BLAST database is built. For genomes with the same taxonomy ID, we arbitrarily select the one with the most recent annotation date, since such genomes are likely to be very similar to each other. Table 4.1 shows an example of several query genomes and their respective clades, as well as the number of genomes in the constructed BLAST database.

Table 4.1: Example clades used to acquire target genomes for a given set of query genomes, and the total number of selected genomes in each clade.

Genome	Clade	Number of genomes in clade
E. coli	*Enterobacterales*	6,311
H. salinarum	*Archaea*	1,125
M. tuberculosis	*Actinobacteria*	8,097
Flavobacteria	*FCB group*	3,306
R. denitrificans	*Alphaproteobacteria*	4,720

4.5.2 Genes with experimentally verified starts

As an initial measure of gene-start accuracy, we use starts that have been verified by N-terminal sequencing. These come from the bacteria *Escherichia coli* [45, 65], *Mycobacterium tuberculosis* [46], and *Roseobacter denitrificans* [66], and archaea *Halobacterium salinarum* and *Natronomonas pharaonis* [48].

4.5.3 Query genomes beyond the verified set

The second set of experiments use a larger number of query genomes randomly selected from four different clades: *Actinobacteria*, *Archaea*, *Enterobacterales*, and *FCB group*. The number of selected genomes per clade is shown in Table 4.4.

This selection of clades was inspired by the grouping of translation-initiation mechanisms described earlier [58]. The archaeal species are known to have large numbers of genes with leaderless transcription. In this case, the upstream sequence exhibits a promoter site pattern close to the gene-start [58]. As for bacterial genomes, they are split into three separate clades. *Actinobacteria* species generally have high-GC genomes with a significant number of leaderless transcripts [58]. On the other hand, *Enterobacterales* genomes populate the mid-GC range, and most almost exclusively feature genes with an RBS-based translation initiation mechanism. Finally, *FCB group* genomes span the low-to-mid-GC range and use a non-canonical RBS.

Table 4.2: The error rates of combining GeneMarkS-2 and StartLink predictions compared to the error rates of the standalone tools, on the set of genes with verified starts.

Genome	Verified	StartLink	GeneMarkS-2	StartLink+
E. coli	769	4.45	3.00	**0.83**
H. salinarum	530	2.73	1.32	**0.43**
M. tuberculosis	701	6.86	9.60	**1.32**
N. pharaonis	315	2.11	0.95	**0.00**
R. denitrificans	526	4.81	3.43	**0.45**
Average		4.19	3.66	**0.61**

Table 4.3: The coverage rates of combining GeneMarkS-2 and StartLink predictions compared to the standalone tools, on the set of genes with verified starts.

Genome	Verified	StartLink	GeneMarkS-2	StartLink+
E. coli	769	99.35	**99.74**	93.50
H. salinarum	530	90.19	**100.00**	87.74
M. tuberculosis	701	83.88	**99.57**	74.75
N. pharaonis	315	89.52	**100.00**	87.30
R. denitrificans	526	81.18	**99.81**	75.86
Average		88.82	**99.82**	83.83

4.6 Results

4.6.1 Experimentally verified starts

Tables 4.2 and 4.3 show the error and coverage rates of StartLink+, compared to the standalone tools, GeneMarkS-2 and StartLink. It is clear that the error rates of StartLink+ are very low. Particularly striking is the improvement on *M. tuberculosis*. As shown in Table 4.2, GeneMarkS-2 and StartLink have 9.6% and 6.8% error rates, respectively, but that error rate drops to 1.7% for StartLink+. This shows the effect of combining independent predictions, even in difficult cases.

Table 4.4: Number of query genomes selected in each clade, and the number of genes predicted by StartLink+.

Clade	Query Genomes	StartLink+
Actinobacteria	111	227,050
Archaea	109	213,336
Enterobacterales	119	394,558
FCB group	105	228,667
Total	**444**	**1,073,611**

4.6.2 Comparison between PGAP and StartLink+

Table 4.4 shows the number of StartLink+ predictions after running on 394 genomes. It is interesting to compare PGAP's annotation to StartLink+, especially when considering the latter's very low gene-start error rate (Table 4.2).

The percentage of gene-start differences PGAP and StartLink+ is not uniformly distributed amongst clades (Figure 4.6). Particularly in *Actinobacteria* genomes, that difference can reach up to 15% of genes per genome, with an average of around 10%. That difference drops to roughly 4.5% in *FCB group* genomes, and to 3% in *Enterobacterales* genomes. The reasons for this variation could be both clade-specific differences in genome GC contents, as well as clade-specific abundance of leaderless transcription.

Similarly, Figure 4.7 shows the error rate as a function of the genome's GC. We see that high-GC genomes from *Actinobacteria* have the highest difference, which is similar to the behavior seen in Figure 4.1. In the verified set, three genomes have high GC content: *H. salinarum* (65%), *N. pharaonis* (63%), and *M. tuberculosis* (66%), and StartLink+ has a low error rate for each (0.6%, 0.0%, 1.9%, respectively). Notably, GeneMarkS-2 and Prodigal's error rates on *M. tuberculosis* are 9.3% and 11.6%, respectively [58].

4.6.3 Performance per StartLink step

As previously mentioned, StartLink steps are ordered by reliability, and the algorithm stops at the first step where a prediction is made. In this section, I show the performance of each

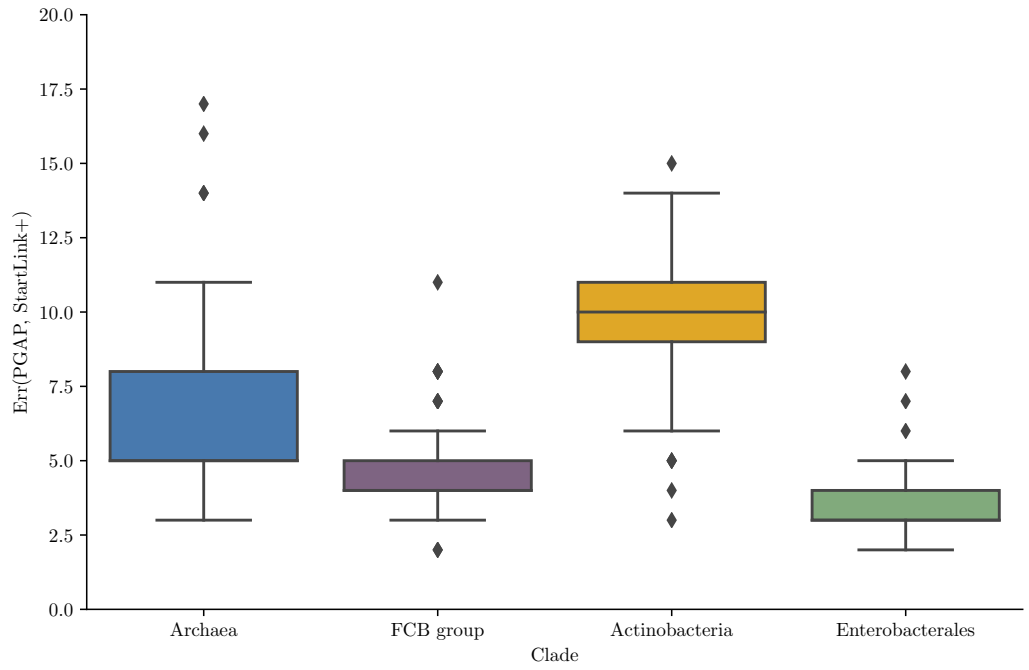


Figure 4.6: Percentages of genes with 5' differences between PGAP and StartLink+ in genomes of different clades.

step, for both StartLink and StartLink+, on the set of genes with verified starts. I then show the distribution of PGAP differences with each step.

On the verified set, the error rate of StartLink's step A is consistently low, close to 0.0% (Figure 4.8, left panel). This is expected, as step A makes predictions under strict conditions and defers more ambiguous cases to the next steps. StartLink's error rate increases slightly after adding steps B and C, but note that StartLink+ maintains a low accuracy overall. The advantage is that the total number of predictions increases dramatically with this addition.

The distribution of differences in gene-start predictions between PGAP and StartLink+'s steps is summarized in Figure 4.9. As expected, the differences are consistently small on step A (in the range of 2-6%) in comparison with steps B and C (5-12%).

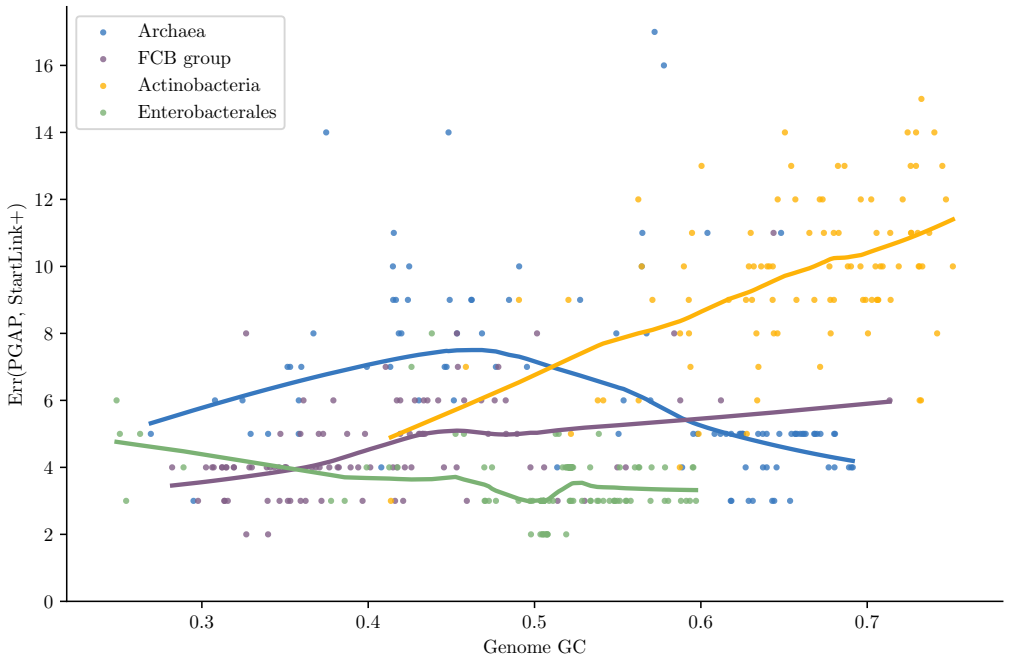


Figure 4.7: The 5' error rate of PGAP compared to StartLink+, as a function of genome GC content.

4.6.4 Conservation of gene overlaps

As previously mentioned, RBS can reassemble more efficiently after translating a gene if the next gene is close by [64]. This fact raised the following question: are gene-overlaps conserved across orthologs? Given all the query and ortholog data along with the MSA, I had just enough information to attempt an answer to this question.

I found that overlaps and short, e.g. $<$10 *nt*, intergenic regions are more likely to be conserved than longer intergenic regions. This is formalized below.

Let us define a *component* to be a query and its selected orthologs, all which are included in the MSA. We restrict this analysis to components with at least $N \geq 10$ sequences. Let $D(n)$ be the nucleotide distance of sequence n from the start of a gene to the end of its upstream gene, and let x be the most frequent $D(n)$ observed in the component. Then, the consistency of upstream genes being x nucleotides away is defined as the **distance**

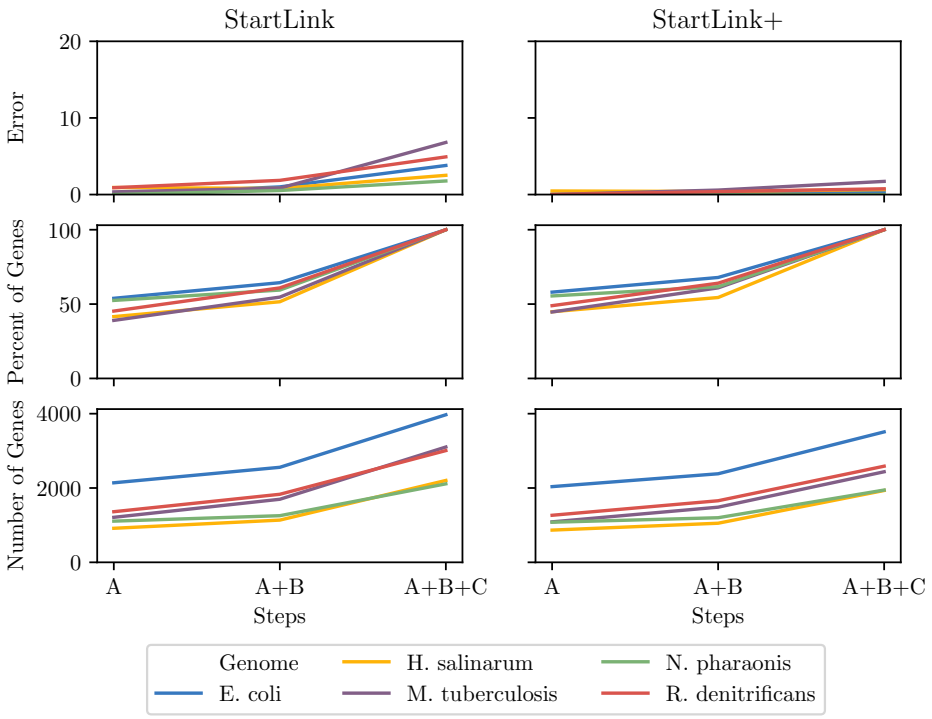


Figure 4.8: Left: The gene-start prediction error rate of StartLink for each step (A, B, C) on the set of verified gene-starts (top), and the percentage (middle) and number (bottom) of StartLink genes predicted by step A alone, steps A and B, and all steps together. Right: Same types of results, for StartLink+.

conservation CC, where

$$\mathrm{CC}(x, f) = \frac{1}{N} \sum_{n=1}^{N} I\left\{x - f \leq D(n) \leq x + f\right\} \tag{4.11}$$

Here, $I\cdot$ is the indicator function and the flexibility f is a characteristic of the stringency of conservation. CC can be interpreted as the probability that the upstream gene for any sequence within a component is located $x \pm f$ *nt* away, where x is the most frequent distance in that component. Figure 4.10 shows that the distance conservation depends on x, and that gene overlaps, $x < 0$, tend to be more conserved.

Zooming in to the -10 to +10 *nt* distances, Figure 4.11 shows the frequency of components for each value of x. By far, most components within that range have $x = -4$, corresponding to a 4 *nt* gene overlap, followed by $x = -1$. This tendency is particularly

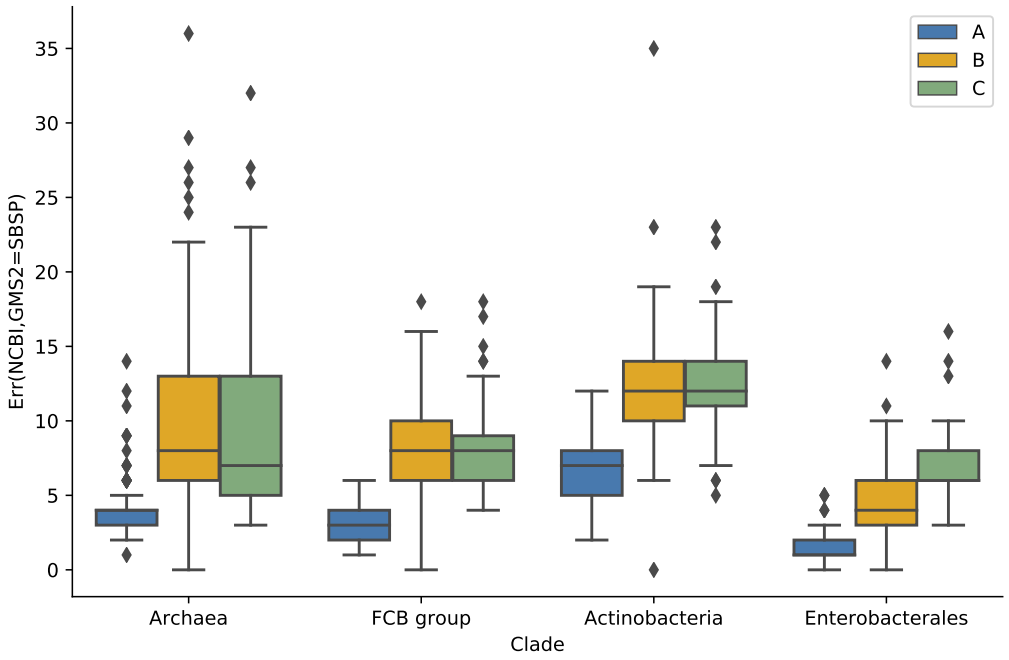


Figure 4.9: The 5' error rate of PGAP compared to StartLink+, shown per step of StartLink.

pronounced in *Actinobacteria*, where more than 60% of components in that range have $x = -4$.

In *FCB group*, components with $x = -4$ constitute only 20% of all components. The reason for this drop when compared to *Actinobacteria* could be that the AT-rich non-Shine-Dalgarno RBS existing in *FCB group* genomes [58] would be under evolutionary pressure to be in a non-coding region, rather than in the higher GC content region of the upstream gene. Nevertheless, the distinct preference of both -4 and -1 overlaps are in agreement with the work suggesting that gene-starts located close to the 3' ends of the upstream genes are favored in evolution [64].

4.6.5 Analysis of distributions of Kimura distances

The behavior of StartLink is clade-specific, and this is likely to be the case for other methods that rely on clade-specific groups of homologous proteins. In this section, I analyze distributions of Kimura distances between queries and their targets across different clades

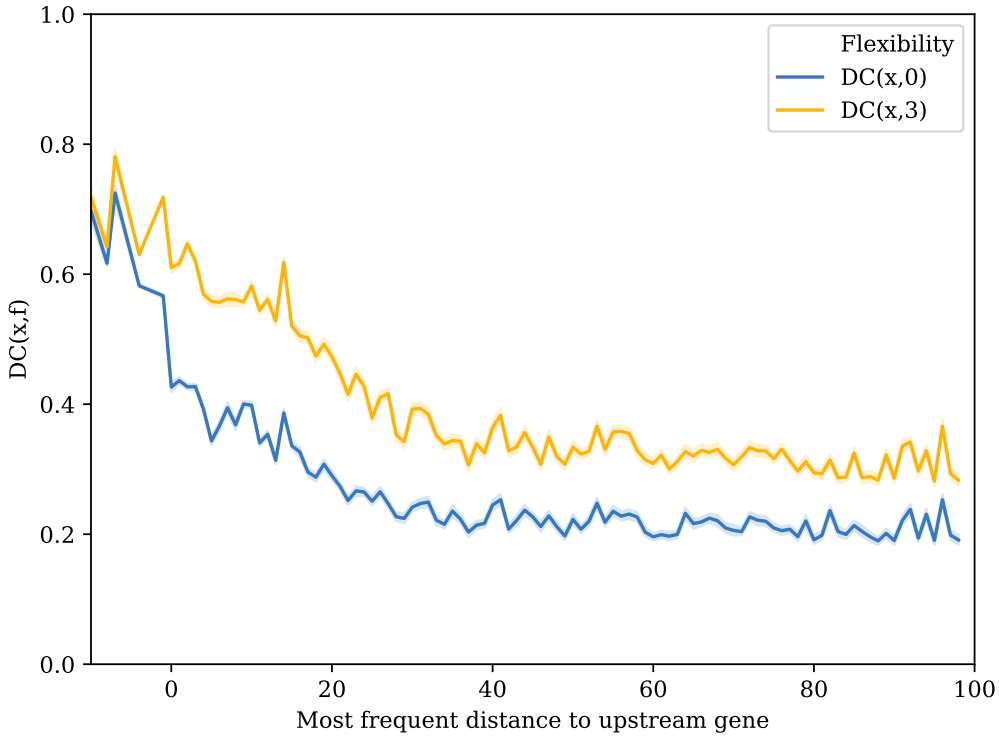


Figure 4.10: The distance conservation as a function of the most frequent upstream distance per component. The data is computed from the orthologs selected by StartLink on 394 query genomes, using PGAP annotation as the positions of gene-starts.

in order to understand better the differences in genomic data available for each clade.

Figure 4.12 shows the distribution of query-to-ortholog Kimura distances, showing contour plots of the frequency of queries that have a given minimum and maximum Kimura distance to their orthologs. The plots indicate a striking contrast in the distributions of the evolutionary distance between queries and targets in each clade. For example, most query genes in *Enterobacterales* have targets both at small (0.1) and large (0.5) Kimura distances. In *Actinobacteria*, however, a large proportion of query genes have their closest relative between 0.25 and 0.5 Kimura. Furthermore, the average Kimura distance per query is 0.38 for *Actinobacteria* and *FCB group* compared to 0.23 for *Enterobacterales* (Figure 4.13).

Thus, we see that orthologs of *Enterobacterales* queries span a broader and more uniform range of Kimura distances; this configuration should produce a more robust performance of StartLink with higher coverage rate per genome. That said, we found StartLink's

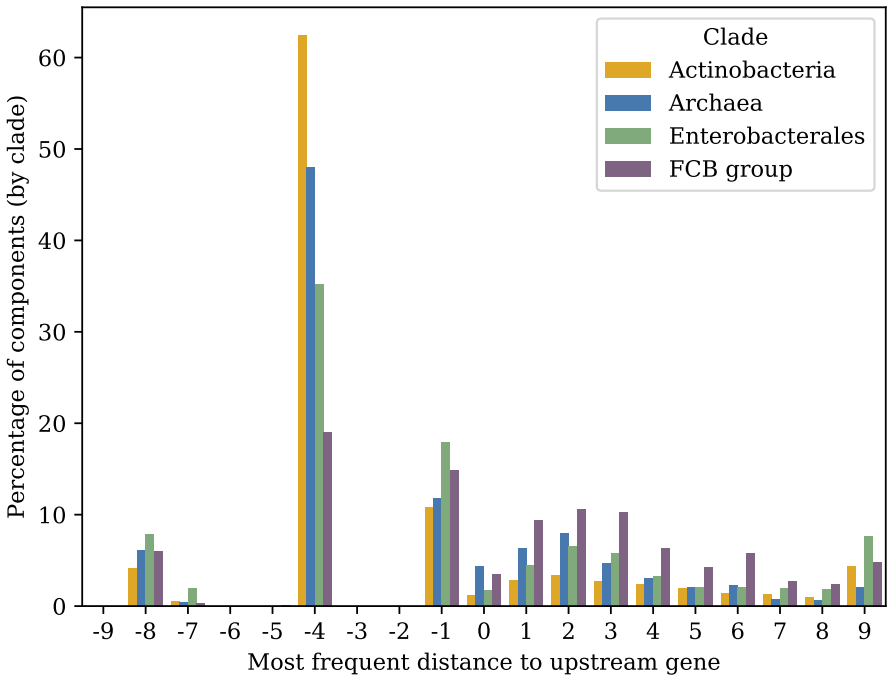


Figure 4.11: The frequency histogram of MSA genomic components with respect to the most frequent distance between same-strand genes. Components in this figure have their most frequent distance x between -10 and +10 *nt*.

and especially StartLink+'s error rates on the verified set to be stable across Kimura ranges (Appendix B.2). We also observed a stable behavior of StartLink+ when comparing its predictions with PGAP's (Figure 4.14).

Regardless of the nature of the differences in Kimura distance distributions, caused by the variability of the speed of evolution or inhomogeneity in the database population, similarity-based approaches in general, and StartLink in particular, have to work in a non-uniform space of orthologs available across multiple clades (Appendix B.2).

4.6.6 BLAST hits across different clades

Besides the variability in Kimura distance distributions, there is also variability across the clades in the numbers of orthologs detected by similarity searches. Figure 4.15a shows the distribution of the number of BLASTp hits prior to any filtering in each of the clades,

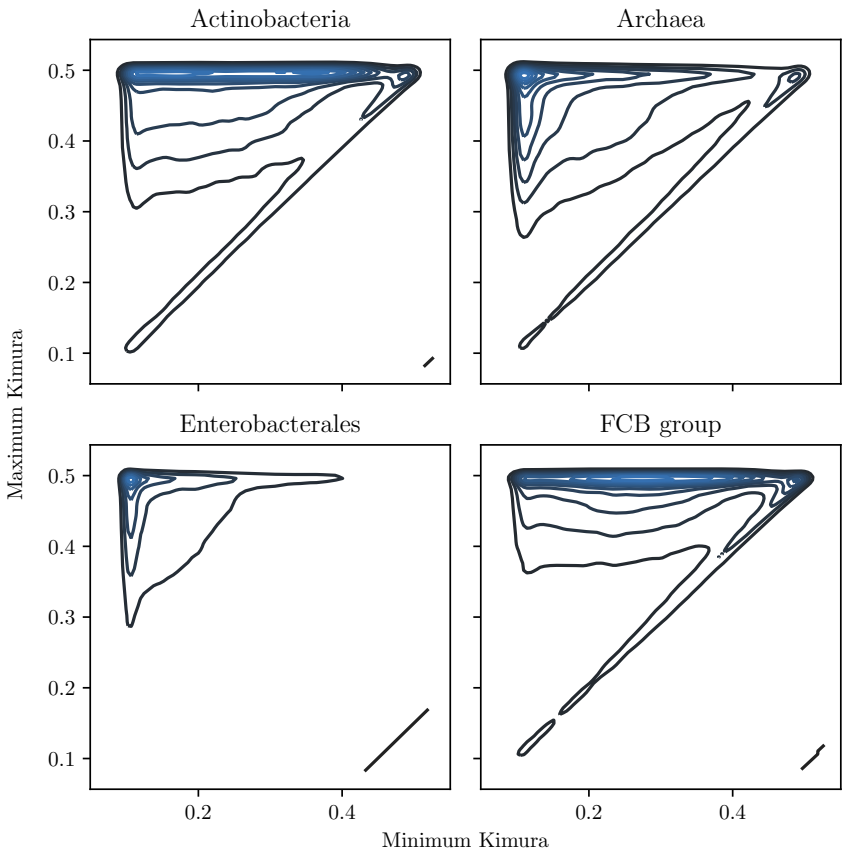


Figure 4.12: The distribution of queries by minimum and maximum Kimura distance to their orthologs. This shows that most query genes in *Enterobacterales* will find orthologs that spread the range from 0.1 to 0.5 Kimura, whereas many in *Actinobacteria* have a minimum Kimura distance of above 0.3 and even 0.4. This is conducted over ~394 query genomes (Table 4.1), and with a total of 1,000,000+ query genes.

and Figure 4.15b shows the percentage of query genes per genome that have at least N BLASTp hits, where N varies from 0 to 5,000 hits.

Naturally, the number of hits per query is directly related to the number of genomes within a clade (Table 4.1). This is easy to see in Figure 4.15b for *Archaea* (1,125 genomes) and *FCB group* (3,306 genomes), where the cumulative distributions rise very quickly and plateau early on, first for *Archaea* and then the *FCB group*. We also see an interesting pattern of behavior in *Enterobacterales* (6,311 genomes) and *Actinobacteria* (8,097). While both of their cumulative distributions grow much more slowly than the first two clades,

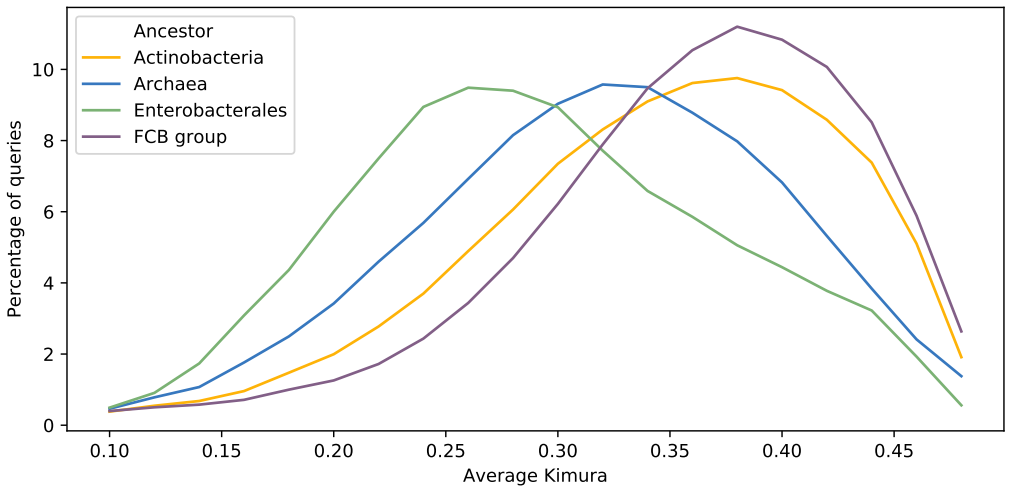


Figure 4.13: The distribution of average Kimura distances (per component). The y-axis shows the percentage of queries (and thus, components) that have a particular average Kimura distance to its orthologs.

Actinobacteria's distribution grows significantly faster than Enterobacterales (which has *fewer* genomes). For example, the likelihood that a query in Enterobacterales gets *at least* 1,000 BLAST hits is $\approx 83\%$, compared to only 60% in *Actinobacteria*.

4.7 Discussion

4.7.1 Comparing StartLink+ and PGAP per step

While StartLink+ has a uniformly low error rate on the verified set of genes across all its steps (A-C), the difference between PGAP and StartLink+ predictions is not uniform (Figures 4.8 and 4.9).

Specifically, there are fewer PGAP differences with gene-starts predicted by step A compared to those predicted in steps B and C. The reasons for this are not immediately clear. It is possible that starts at LORFs, which are targeted by step A, are easier to predict. This is because if the region between the candidate start at the LORF and the next candidate start is determined to be a coding region, then this automatically leaves a single candidate start for selection. Furthermore, detecting a block of conservation is much more reliable

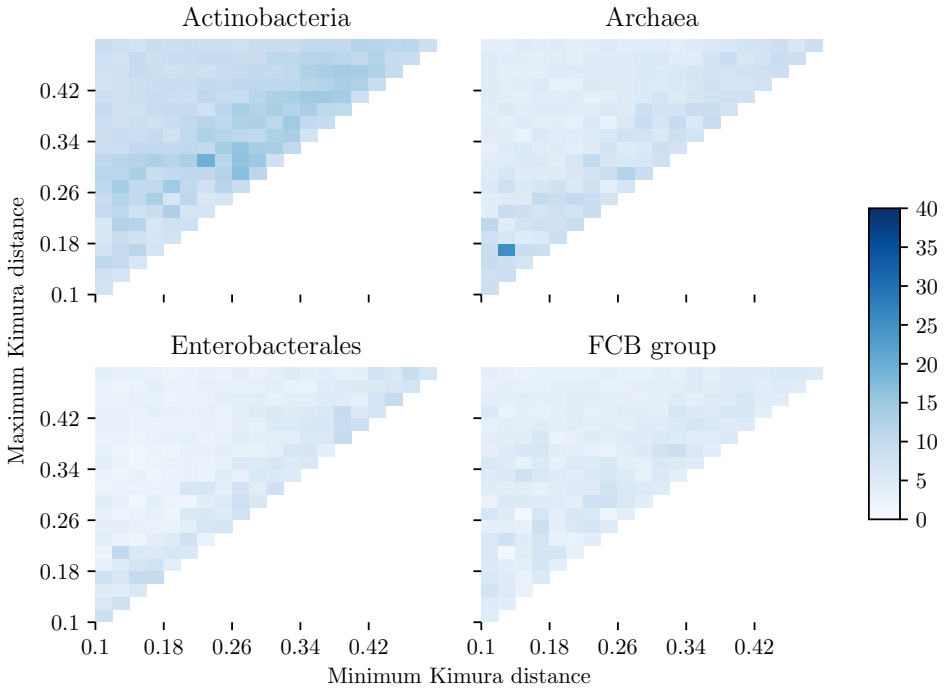


Figure 4.14: The percentage of gene-start mismatches between PGAP and StartLink+ (computed as Err (PGAP,StartLink+)) as a function of the minimum and maximum Kimura distances between a query and its targets. The color bar encodes the error rate. The analysis is on the same data mentioned in Figure 4.12.

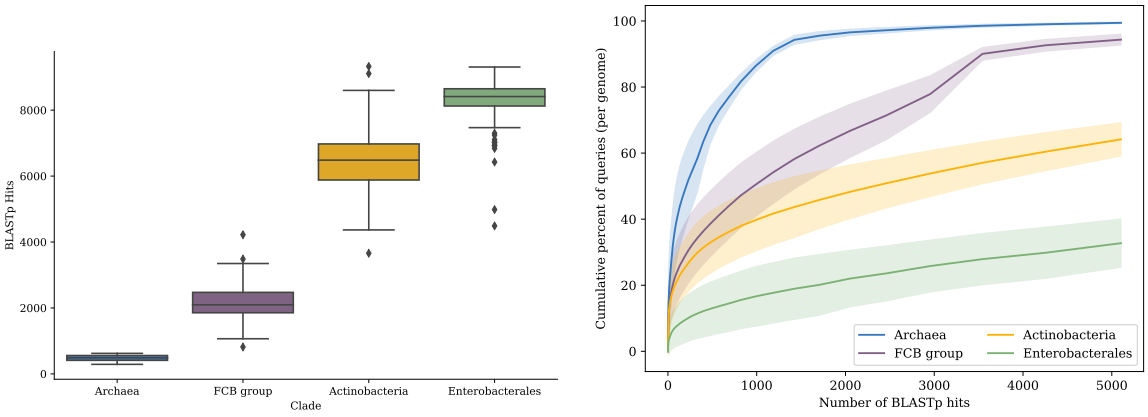
than detecting the conservation of a single amino acid, which is done in steps B and C.

Most of PGAP's differences with StartLink+ comes from steps B and C. Given that StartLink+ maintains a low error rate per step on the verified set, this suggests that PGAP annotation could be improved by taking StartLink+ predictions as reference points.

4.7.2 Coverage of StartLink and StartLink+

As a standalone tool, StartLink achieves an 85% coverage rate on average (Figure 4.16a). In particular, it achieves a rate of 92% in *Enterobacterales*, significantly higher than that of the remaining three clades, all of whose averages lie in the 80-83% range.

The coverage of StartLink depends heavily on the available sequences in the database. Specifically, observing the number of initial, unfiltered BLASTp hits gives an upper bound

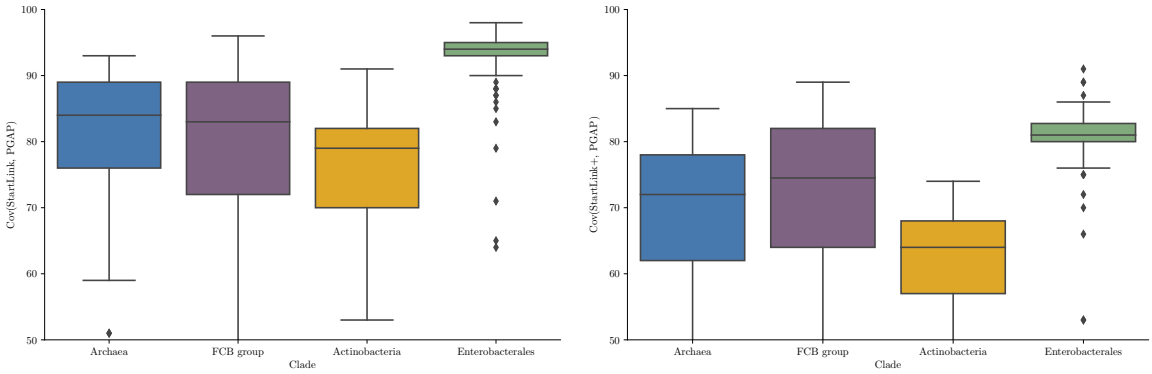


(a) (b)

Figure 4.15: Distribution of raw blast hits across clades for the set of query genomes in Table 4.1. (a) Box plots for the raw number of BLASTp hits per clade. (b) The cumulative percentage of queries for a given clade with *at most* N BLASTp hits, where N varies from 0 to 5,000. The shaded bands show the standard deviations (per clade) across query genomes.

on the possible coverage. Figure 4.17 shows the cumulative percentage of queries per genome that have at most n BLASTp hits. Here, we consider $n \in [0, 40]$ since we are interested in queries with little to no BLAST hits. Note that most queries generally have hundreds or thousands of hits (Figure 4.15b). Interestingly we see that on average, 10% of *Archaea*, 12% of *Actinobacteria*, and 12% of *FCB group* genes per genome have *fewer* than 10 BLASTp hits. This is compared to only 3% for *Enterobacterales* genes. Further note that these hits may not fit within the desired Kimura distance to the query and to each other. If we compare this to the coverage rates of StartLink for each of the clades, we see that a big percentage of the loss of coverage can be traced back to the low number of hits.

The differences between the clades are larger for StartLink+, which has an average coverage rate of 73% (Figure 4.16b). Interestingly, *Actinobacteria*'s coverage drops by 16% from StartLink to StartLink+, which is the biggest drop across the clades. This drop measures the disagreement of gene-starts between GeneMarkS-2 and StartLink; it represents the number of genes that StartLink+ filters out in order to maintain a very low error rate, as shown in Table 4.2.



(a) (b)

Figure 4.16: Coverage rates of StartLink and StartLink+ shown for different clades. The analysis is on the same data mentioned in Figure 4.12.

4.7.3 Differences in numbers of selected orthologs

The maximum number of allowed targets N, set to 50, is used to limit the number of sequences in the MSA. This is done by selecting at most N sequences from the large number of BLAST hits, and running them through an additional filtering process (see Appendix B.5).

As such, the effective number of orthologs per query at the end of a StartLink run can be less than N. Figure 4.18 shows the average number of targets per query after a full StartLink run. We see that the number of targets per query can differ significantly across clades, especially when comparing *Archaea* and *Enterobacterales*.

One reason why *Enterobacterales* has a high average is possibly due to the large spread of Kimura distances compared to other clades (Figure 4.12). Comparatively, the spread of averages within *Archaea* is non-uniform, reaching as low as ~10 targets per query on average. This is partly due to the sparseness and small number of genomes in this clade, making it statistically less likely that we find enough sequences within the right Kimura range.

With all that said, note that the difference in the final number of targets per query observed does not translate into a large difference in performance on the verified set. For

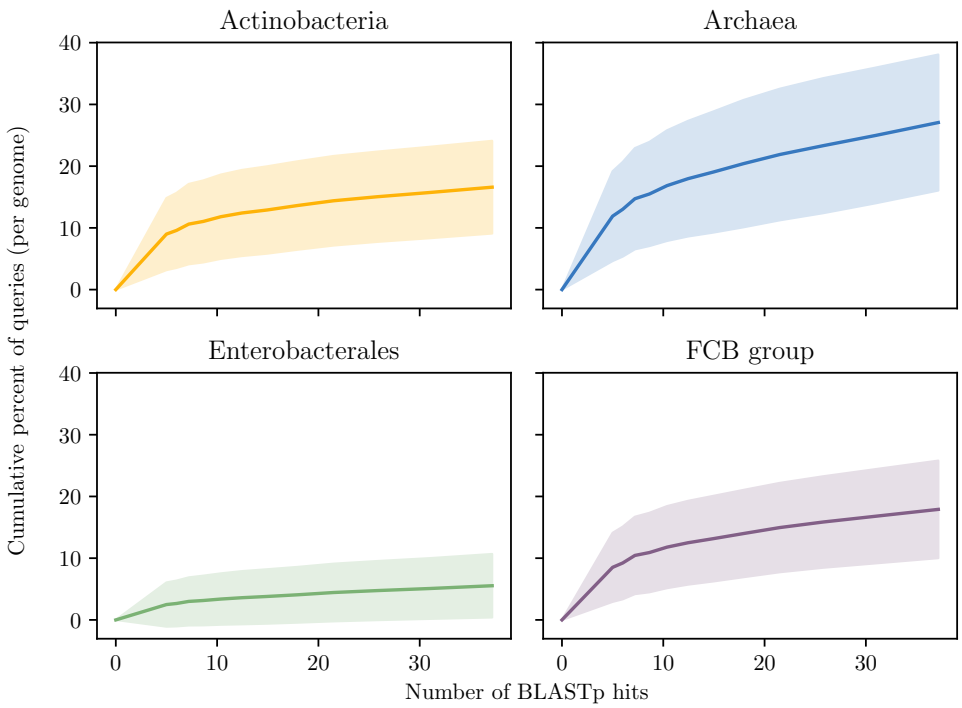


Figure 4.17: The cumulative distribution of BLASTp hits (¡ 40) per query (in a genome), shown for different clades. This is a zoomed in version of Figure 4.15b

example, when StartLink is run with $N = 50$, both archaea, *H. salinarum* and *N. pharaonis*, end up with 20 targets per query on average, compared to *E. coli*'s 40. In terms of error rate, however, *H. salinarum* and *N. pharaonis* make an error of 3% and 2% respectively, while *E. coli*'s error rate is 5%.

We can set N to 20 to further test out whether 20 targets affects StartLink's performance. This resulted in 10 to 15 targets per query for both archaea, but there was no real change in error rate; there is a slight increase in error rate for *H. salinarum* by 0.6% and a decrease in error rate for *N. pharaonis* by 0.7%. In fact, on all verified genomes, we saw a 0.5% change in error rate on average when N was shifted, some positive and some negative. This shows that StartLink seems to be stable with respect to N, and these minor shifts can be attributed to the underlying randomness of selecting the target sequences.

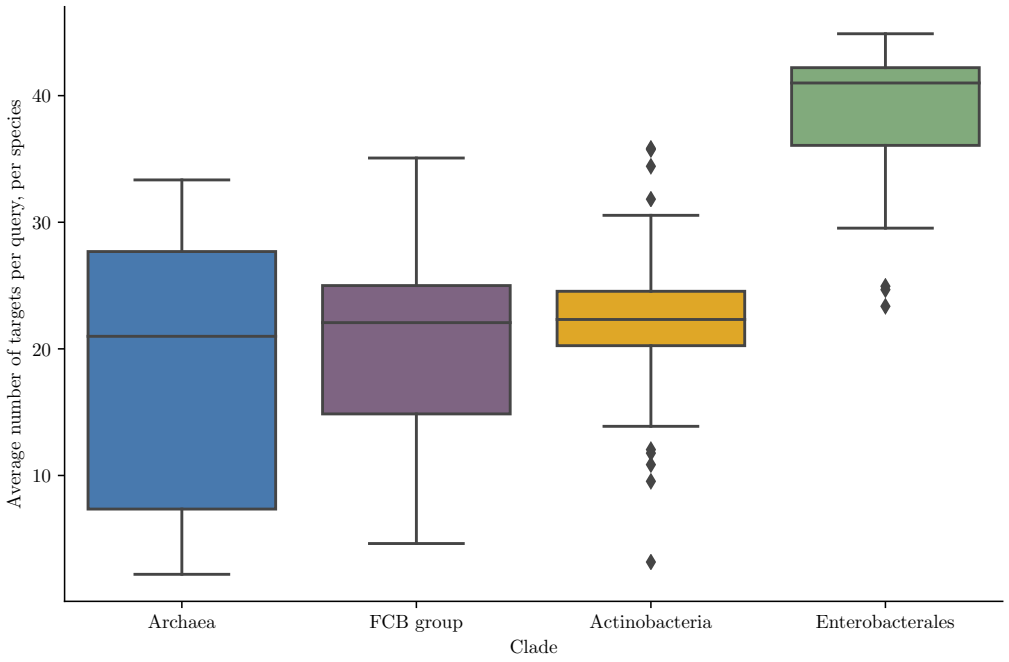


Figure 4.18: The average number of targets per query at the end of an StartLink run. The average is computed per genome, and shown for each of the four clades on the set of 400 query genomes.

4.7.4 Example Alignments

In addition to the locations of the predicted genes, StartLink can output the multiple sequence alignments used to make these predictions. Figure 4.19 shows an example of an MSA where StartLink+ disagrees with PGAP's start location.

The example shown is from the archaea *Haloferax sp.*. The first amino acid sequence (line 4) is the translated query sequence, following which are the selected orthologs. Capital M, V, and L letters represent Methionine, Valine, and Leucine amino acids coded by ATG, GTG, and TTG respectively. Non-capital letters are amino acids encoded by other start codons; specifically, small-case v and l represent valine and leucine amino acids coded by non-GTG or TTG codons, respectively.

PGAP's prediction, represented by "#ref", chooses a GTG-coded valine amino acid, while StartLink+, represented by "#selected", chooses a start upstream of it. In this case, two factors indicate why StartLink+'s selection is more likely to be correct. First, the region

```
#selected          M-------------------------------------------------
#q-3prime          --------------------------------------------------
#ref               ----------M---------------------------------------
215415;215993;-;A Mrrtvfvaatvlalltagvaaaflagvgpfadtts-adgsdgafptqtta
1;0.1456           Mrrtllvaatvlvlltagisaafltgvgpfadtts-addsdgafptqtta
2;0.146            Mrrtllvaatvlvlltagisaafvtgvgpfsddsa--aesdeafptqtta
3;0.164            Mrhpllaaatvlallttgvvaafvtgvgpfadttp-agdsdgafptqtta
4;0.1727           Mrrtllvaatvlvlltagisaafvtgvgpfsddsa--aesdeafptqtta
5;0.1751           Mrrtllvaatvlvlltagvsaafvtgigpfsdda---gdsdepfptqtta
6;0.1773           Mrrtllvaatvlvlltagvsaafvtgigpfsdda---gdsdepfptqtta
7;0.1783           Mrrtllvaatvlvlltagvsaafvtgigpfsdda---gdsdepfptqtta
8;0.1803           Mrrtllvaatvlvlltagisaafvtgvgpfsdds---aesdeafptqtta
9;0.1862           Mrrtllvaatvlvlltagvsaafvtgvgpfsddd---aesdepfptktta
10;0.2057          Mrrtllvaatvlvlltagvsaafvtgvgpfsddd---aesdepfptktta
11;0.227           Mrrtllvaatvlvlltagvsaafvtgigpfsddd---adsdepfptqtta
12;0.2291          Mrrtllvaatvlvlltagvsaafvtgigpfsddd---adsdepfptqtta
13;0.2293          Mrrtllvaatvlvlltagvsaafvtgigpfsddd---adsdepfptqtta
14;0.2316          Mrrtllvaatvlvlltagvsaafvtgigpfsddd---adsdepfptqtta
15;0.3906          Mkrsllvasavvllvaagvgvsfvtgigpfadgsdagdqsadpfptqtat
16;0.3976          Mkqstlvvlalvalvgsgvgaafvtgvgpfasddd---eldgefptqtat
17;0.4102          Mkrstlvvlavlalvgggvatafvtgvgpfasddd---eldgefptqtat
18;0.4142          Mkqstlvvlalvalvgggvatafvtgvgpfasddd---eldgefptqtat
19;0.4291          Mrrpllavvavllvvgggvtvafatgfgpfaggadssdqptepfptqtpt
20;0.4331          Mkrsllvastvvllvaagvgvafvagigpfadgsdagdqstdpfptqtat
```

Figure 4.19: *Haloferax sp., Archaea*

between the two starts is highly conserved, even though the Kimura distances between targets and query extends to the full allowed range. Such a high conservation implies that this region is very unlikely to be a non-coding region, as shown in Figure B.4. Second, while almost all target sequences also have a Valine amino acid at the position selected by PGAP, all of them are non-GTG (small letters), meaning that they cannot be a valid gene-start. Comparatively, StartLink+'s position is made entirely of ATG codons across orthologs. More examples are shown in Figure B.8.

4.8 Conclusions

In this work, I showed that existing gene-start prediction tools predict conflicting starts for 15-25% of genes, despite seemingly highly accurate predictions on the small sets of genes with verified starts. This performance does not generalize to a large number of genes and genomes, especially those with high GC content.

To address this, I showed how combining algorithms with independent information sources can filter low certainty predictions thereby decreasing error rate of the remaining

set. This was done using StartLink+, an approach that combines the comparative method StartLink with the *ab initio* predictor GeneMarkS-2 to filter predictions with low-certainty, reducing the gene-start error rate to $<2\%$. In future work, the sets of gene starts predicted by StartLink+ can be used to learn sequence patterns around gene-starts in more details, and facilitate further in-depth understanding of the biological diversity of translation-initiation mechanisms. This work is currently submitted to BioRxiv under the title "StartLink+: Prediction of Gene Starts in Prokaryotic Genomes by an Algorithm Integrating Independent Sources of Evidence."

CHAPTER 5

METAGENEMARKS

5.1 Introduction

With standard genomics, we are able to bring individual species into a lab environment for cultivation and sequence their DNA. This, it turns out, is not the case for most microbial species. A 2007 study estimated that only 0.1-1% of soil-inhabiting bacteria can be cultured in a lab, with an even lower fraction for bacteria from aquatic environments [67].

In such cases, instead of isolating individual species for further inspection, **metagenomics** allows us to inspect the environmental sample as a whole. Short DNA fragments, called reads, are extracted from a sample containing multiple species. The fragments belonging to the same species are assembled together, forming a longer strand of DNA.[1] This leads to a set of longer DNA fragments, where each fragment comes from a single species. The steps are shown in Figure 5.1.

Unfortunately, we are not always able to recover the full or even most of the genome for each species, and some species may very well be over or under represented. Nevertheless, metagenomics provides a way of studying species in a microbiome sample.

From a gene-annotation point of view, this poses its own restrictions. The small size of many individual fragments and the fact that they come from diverse sets of species means that we cannot easily build a single predictive models as was done in GeneMarkS-2, unless that model can somehow account for this diversity. This chapter presents MetaGeneMarkS, an approach that builds a set of models in an attempt to represent the wide range of prokaryotic species. Based on MetaGeneMark, MetaGeneMarkS extends its predecessor by im-

[1] The details of this step are beyond the scope of this work. That said, it should be noted that the no prior knowledge of the species is required for successful assembly, although some approaches may use *a priory* knowledge if available for increased sensitivity.

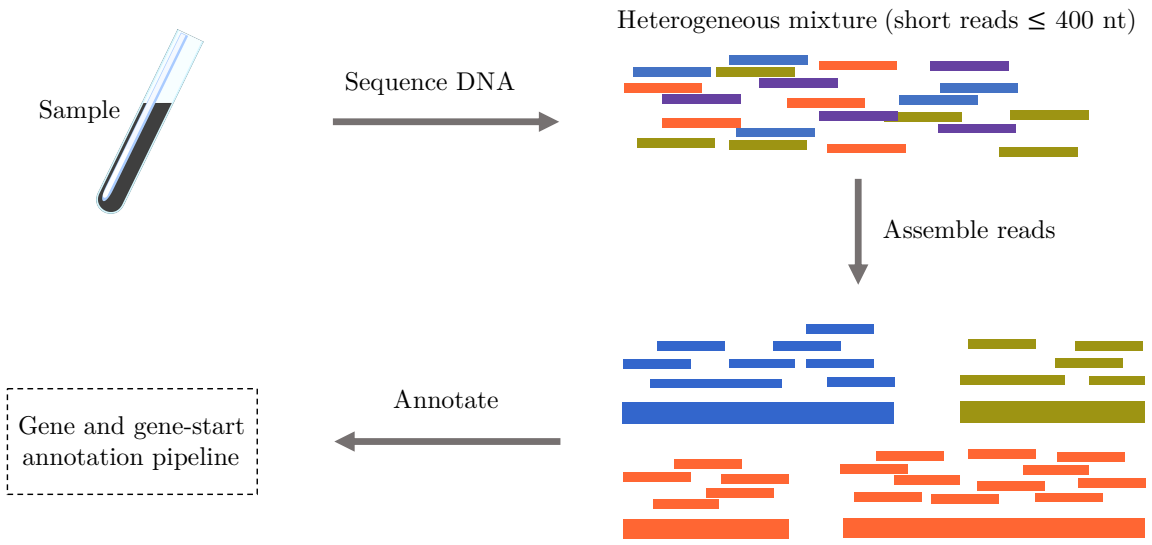


Figure 5.1: An illustration showing the steps to build a metagenome from a biological sample.

proving its gene-start prediction accuracy[2]; specifically, it incorporates RBS, promoters, and other gene-start related components from GeneMarkS-2 into the metagenomic case.

5.1.1 Goldilocks and the three-thousand microbiome species

Consider a single, fully-sequenced genome. In such cases, building single gene and gene-start models allows for accurate prediction of most genes[3] (Chapter 3). This is because most genes present in a single genome have, through their evolutionary history, been exposed to similar internal and environmental forces, which ultimately has an effect on their compositional and structural bias.

The problem in metagenomics is that we are often faced with short fragments coming from a heterogeneous set of species, making it neither likely that a single model can cover all species due to their wild heterogeneity, nor that a single model can be dynamically built for each individual short fragment, as is done in GeneMarkS-2. Furthermore, comparative approaches, such as those shown in Chapter 4, will fail when encountering rare or

[2]In actuality, MetaGeneMarkS extends an unpublished version called MetaGeneMark2; this latter uses a more nuanced criteria to select which models are best suited for finding genes in the current sequence. See Appendix C for more details.

[3]I say "most" because GeneMarkS-2 also uses MetaGeneMark to account for "atypical" genes.

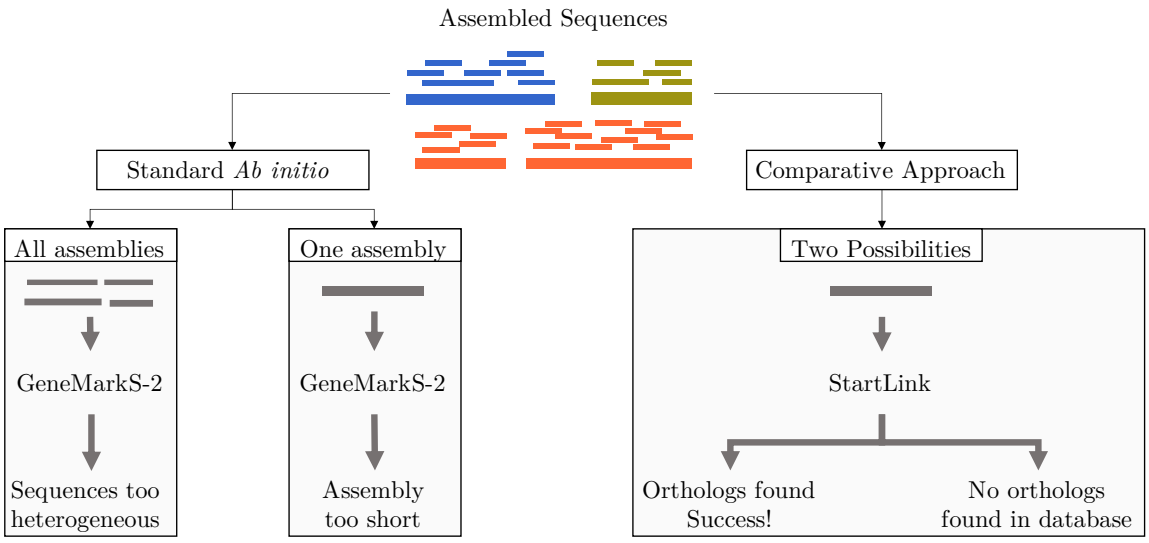


Figure 5.2: An illustration showing why unsupervised *ab initio* and comparative approaches are frequently not suitable for metagenomic prediction.

previously unknown genes, which is expected to be frequent in metagenomes. Figure 5.2 summarizes these difficulties.

In comes Goldilocks: if building a single model to cover all sequences is too simplistic and building a model per sequence is not possible, what lies in between? Intuitively, We can collect "similar" sequences together, and build a separate model for each such set. Conceptually, let S be the entire set of fragments gathered from a microbiome. If we can separate S into non-overlapping subsets of **homogeneous** groups, then each of these subsets, assuming it has enough members, can be represented by its own gene-model. Then, to annotate a new fragment, we simply choose the gene-model that best fits the characteristics of this fragment.

As it turns out, a genome's GC content, along with the type of translation-initiation mechanisms used serve as strong features that allow us to build these homogeneous groups. A study done on 2,670 prokaryotic genomes showed that although species had different phylogenetic lineages, their similar GC contents forced them to adopt the same codon and amino acid usages [68], with similar results observed by other studies [38]. In fact, [68] showed that codon and amino acid usage change linearly with a change in GC.

This idea forms the foundation of MetaGeneMarkS, with one difference. Instead of learning new models for each metagenome, we learn them once over a large set of prokaryotic genomes, and use these models for any new prediction task.

5.2 Related Works

5.2.1 MetaGeneMark

In MetaGeneMark [38], 319 bacterial and 38 archaeal genomes were used to determine parameters of a model for gene-coding regions in each unit-sized GC bin, from 30 to 70. These parameters took the form of a 5^{th} order, three-periodic Markov model, and represented each six-letter word occurring in the coding region. Then, a regression model fit the frequency of each six-letter word to the change in GC content. This is equivalent to constructing a regression of the entire coding model, assuming that the the words are independent. The authors also showed that using GC as the variable allows us to fit a reasonably smooth curve for all parameters.

Apart from being a successful metagenomic gene-finder in its own right, MetaGeneMark has since been deployed in GeneMarkS-2 to account for horizontally-transferred and atypical genes, i.e. genes whose GC composition differs from the bulk of genes within a given genome.

5.2.2 MetaProdigal

MetaProdigal [69] uses 50 pre-selected and pre-trained Prodigal models to find genes in a new DNA sequence. These models were selected from Prodigal runs over 1,400+ different species. The combination of 50 models was selected based on how each combination performed on a test set. For any new fragment, MetaProdigal runs the standard Prodigal algorithm using each of the 50 models, and selects the one that achieves the highest score.

MetaProdigal was shown to perform better than MetaGeneMark in its assignment of gene-starts. This is due to the powerful RBS models built by Prodigal and to the lack of a

gene-start model in MetaGeneMark.

5.2.3 FragGeneScan

FragGeneScan [70] is the only tool from those listed here that incorporates sequencing errors into its model. It uses an HMM that models gene-coding and start/stop codon frequencies. Instead of modeling translation-initiation mechanisms directly, FragGeneScan uses a 63nt positional weight matrix centered around the gene's start codon. The matrix is based on a posterior probability of the start-codon begin true, where the probability is characterized by two fitted Gaussian distributions derived from true and false start codons.

5.2.4 MetaGeneAnnotator

MetaGeneAnnotator [71] uses the concept of an "RBS-map" to model ribosomal binding sites. Essentially, an RBS-map tracks RBS models along with their relative position from the start codon; this is to allow for motifs being located at slightly different positions from the gene-start in different species. MetaGeneAnnotator builds this map by analyzing candidate motifs based on their match to a pre-determined 16S rRNA tail: G(A/T)(A/T)AGGAGGT(G/A)ATC. While this provides a solid structure for species that use the standard Shine-Dalgarno RBS, it fails to account for non-canonical RBS and for cases of leaderless transcription. As shown in Chapter 3, these two cases were significant in almost 40% of genomes analyzed from the set of 5,007 representative genomes.

5.3 Methods

In this section, I describe how we can learn GC-dependent models for four types of GeneMarkS-2 models: RBS/promoter motifs and spacer-length distributions, start codon probabilities, and start context positional Markov models. Furthermore, separate versions of each will be learnt for the different GeneMarkS-2 groups, i.e. A, B, C, D, and X (see Chapter 5 for a detailed explanation of these groups).

The methods used for start-codon and start context models are based on a non-parametric regression in the probability space as a function of GC, similar to that used in MetaGeneMark to estimate parameters of coding regions. On the other hand, the motif models require a more complex procedure to account for variability in size and shape of RBS and promoter motifs.

5.3.1 Notation and Setup

Let $\mathcal{A}$ and $\mathcal{B}$ be the sets of archaeal and bacterial genomes respectively, with $\mathcal{A}^{(G)}$ and $\mathcal{B}^{(G)}$ indicating the corresponding genomes classified by GeneMarkS-2 into group $G \in \{A, A^*, B, C, D^*, X\}$. The "*" indicates that this group is built from archaea genomes. A lowercase g will be used to identify a particular GC bin as needed (e.g. $g = [30, 31)$ or $g = [40, 45)$).

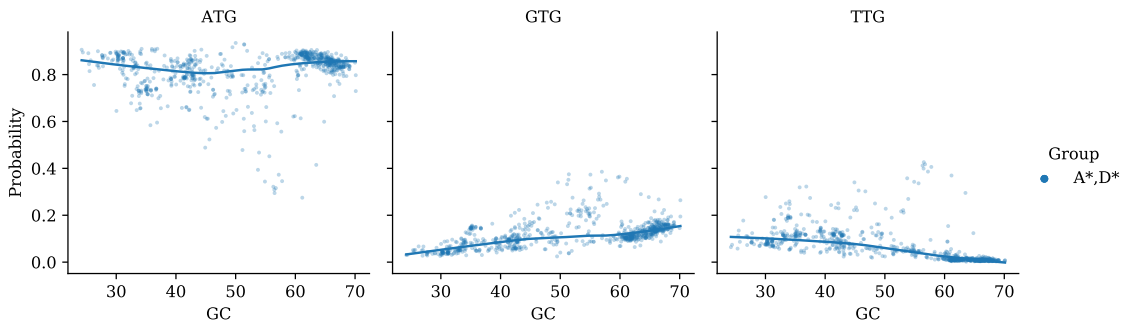
We begin by building a GeneMarkS-2 model for each genome in $\mathcal{A}$ and $\mathcal{B}$. This provides us with individually trained models for each, as well as the group label G assigned by GeneMarkS-2.

5.3.2 GC binning

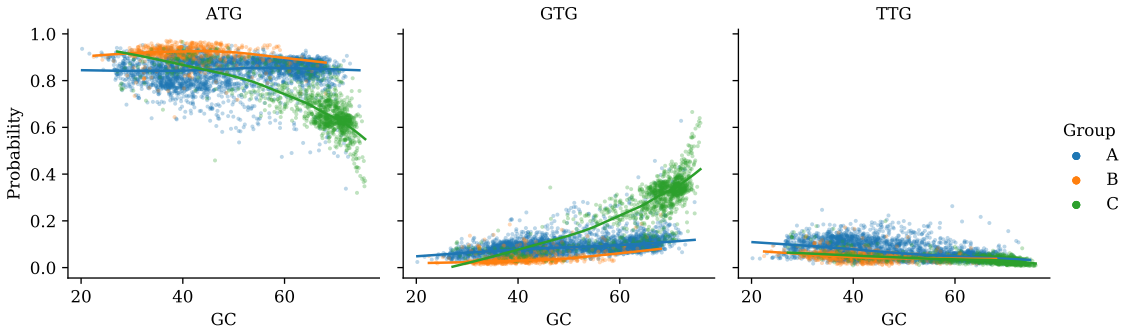
The original version of MetaGeneMark constructs models for each GC bin $g = [g_l, g_u)$, where bins are defined by their lower and upper limits g_l and g_u. It uses 40 unit-sized, non-overlapping GC bins for each of archaea and bacteria, starting from GC content 30 up to 70; i.e. $(30, 31), (31, 32), \ldots, (69, 70)$. The same process is used in MetaGeneMarkS, with the exception that bins for motif models will be of size 5 instead of 1.

5.3.3 Start Codon Probabilities

Let p_s, where $s \in \{\text{ATG}, \text{GTG}, \text{TTG}\}$, be the start codon frequencies we get by running GeneMarkS-2 on a given genome. Figures 5.3a and 5.3b show the probabilities of start codons as a function of GC for genomes from $\mathcal{A}$ and $\mathcal{B}$. We can see clear trends for how



(a) Archaea



(b) Bacteria

Figure 5.3: The probabilities of start codons derived by GeneMarkS-2 runs over the set of archaeal and bacteria genomes. Colors represent the corresponding GeneMarkS-2 groups for each genome.

the probabilities change across GC, especially when broken down by group.

With this set up, we fit a locally weighted scatterplot smoothing (LOWESS) regression to datapoints for each start codon, per group. Note that since datapoints for archaea are scarce especially when considering high-GC group A genomes, a single regression model is fitted to all archaea genomes.

Following that, each regression is "discretized" by computing its average per GC bin. For each group G in archaea and bacteria, we end up with a single probability value for each of ATG, GTG, and TTG for every GC bin g. Formally, the probability of codon s for group G genomes in GC bin g is defined as

$$p(s; g, G) = \frac{1}{Z_{g,G}} \operatorname{avg}\left(L_{g,G}(p_s)\right) \tag{5.1}$$

wheer $L_{g,G}(p_s)$ is the set of values obtained by the LOWESS fitting of p_s values from group G in GC bin g, avg is the mean function, and Z is a normalization constant to ensure the end result is a valid probability distribution of s, g, and G.

5.3.4 Start Context Models

The start-context model used in GeneMarkS-2 is a second order positional Markov model, located at positions -3 to 12 nt relative to the gene-start. Its parameters consist of a probability value per three-letter word per position, i.e. AAA, AAC, AAG, . . . , TTT. The regression fitting process is similar to that of start codon probabilities, except that it is done for each word at each position.

5.3.5 Motifs and Spacers

Constructing GC-dependent models for RBS and promoter motifs and spacers requires more preparation. For clarity, I limit this explanation to RBS models, noting that the process for building promoter models works in much the same way.

It is not straightforward to compare RBS models across different genomes. This is in part due to: (1) shifted models, e.g. AGGAGG vs AAGGAG, (2) fluctuations in probabilities per position, and (3) the fact that some genomes may favor shorter RBS, e.g. GGAG, while others prefer longer, e.g. AGGAGG. In particular, note that RBS models in GeneMarkS-2 have a fixed width of 6 *nt*, meaning that GGAG would be represented by a consensus of the form NNGGAG, NGGAGN, or GGAGNN, where each N can be any nucleotide. This makes the combination per position, as was done for the start context model, not straightforward.

In the following two parts, I describe the process of building GC-dependent motif models and their spacers, and how they can be used in the prediction step.

5.3.5.1 Intuition

Suppose we are given four RBS models, the motifs' positional Markov models and their spacer distributions, with the following consensus sequences:

```
A:    AAGGAG
B:    CAGGAG
C:    AGGAGG
D:    CGGAGG
```

Instead of combining all models together, we separate the models into clusters based on how similar there consensus sequences are. Practically, we find a small number of clusters such that within each cluster, each pair of consensus sequences can differ by at most two nucleotides.

In the above example, notice that A and B have the core AGGAG starting from the second position, while C and D have GGAGG from the second position. Therefore, A and B form one cluster, and C and D for another, and models within the same cluster can be "merged" together by computed the mean probability per letter at each position of the motif. We end up with two merged models representing the "average" motif model of each cluster. This allows us to avoid the problem of merging models that are "shifted" relative to each other.

For each cluster, the spacer distributions corresponding to the models within that cluster are also merged by simply averaging the probabilities at each distance value. We found that spacer distributions can vary significantly even within a type of consensus (e.g. an RBS model with consensus AGGAGG can have an average distance from the gene start of 4nt in one genome, and 8nt in another genome). Therefore, attempting to cluster spacer models by their mean (in combination with the motif clustering above) leads to a large number of different average models, which increases model complexity and runtime. As described in the supplementary materials, a simple average model allows us to model this diversity

without the added complexity.

In the end, for each GC bin and genome group, we end up with a small number of "average" RBS/spacer models (one pair for each cluster), and these will be used together to predict the RBS for a given gene-start. The process is formalized and described in detail below.

5.3.5.2 Training Step: Building GC-dependent motif and spacer models

Let Λ be the entire set of RBS models collected by running GeneMarkS-2 on a set of genomes. I use $\theta = (g, G)$ to represent a specific setting of the GC bin g and group G. Let $\lambda(\theta)$ be the indices of RBS models in Λ from group G genomes in GC bin g; i.e., the set of group G models in that GC bin is defined as $\mathcal{M}_\theta = \{(m_n, s_n); n \in \lambda(\theta)\}$, where m_n and s_n are the motif and spacer models for model n. Consider all $m_n \in \mathcal{M}_\theta$. We extract their consensus sequences and cluster them (allowing at most two differences between each pair of sequences within the same cluster). This gives us the assigned cluster h_n for each motif, and motifs with the same h_n value will be merged together.

For each GC bin, we construct a model with the following components: (1) a cluster prior model, representing the frequency of motifs assigned to some value h, (2) a set of "merged" motif models (positional Markov models) and (3) merged spacer models, one for each h. The cluster prior model allows us (at prediction time) to assign higher weight to one set of models over another. E.g. if AAGGAG (cluster 0) models are much more frequent than AGGAGA (cluster 1) in our training set, then the prior model will automatically assign higher weights to the merged "cluster-0" model.

Cluster Prior Model

Let $H(h; \theta)$ be the set of model indices (from $\lambda(\theta)$) that are assigned to cluster h. Then, the probability of a model being assigned to h is

$$p_H(h; \theta) = \frac{1}{N_\theta} |H(h; \theta)| \tag{5.2}$$

where N_θ is the number of group G models in GC bin g.

Motif Model

For each setting of θ, we compute a set of motif models based on the cluster assignments. Let $M_{\theta,h}$ be the new average motif model representing cluster h. The probability of nucleotide l at position i is determined by

$$M_{\theta,h}(i,l) = p_M(l|i;h,\theta) = \frac{1}{Z_{\theta,h,i}} \sum_{n \in H(h;\theta)} m_n(i,l) \tag{5.3}$$

Here, $Z_{\theta,h,i}$ is the normalization constant for position i of the merged motif model $M_{\theta,h}$, and $m_n(i,l)$ is the probability of letter l in position i in motif model m_n.

Spacer Model

We merge spacer models corresponding to the motif models that are assigned to the same cluster. For a given cluster h, the probability of a motif being d nucleotides away from the gene-start is then computed as

$$S_{\theta,h}(d) = p_S(d;h,\theta) = \frac{1}{Z_{\theta,h}} \sum_{n \in H(h;\theta)} s_n(d) \tag{5.4}$$

where $s_n(d)$ is the probability (determined by model s_n) that a motif exists d nucleotides upstream of the gene-start.

Putting everything together

For any setting of $\theta = (g, G)$, the final model is made up of three major components: the prior probability p_H, and the cluster-indexed sets of motif $M_{\theta,h}$ and spacer $S_{\theta,h}$ models. These are grouped together into $\mathrm{MGM}_\theta = \{p_H, M_{\theta,h}, S_{\theta,h} | h \in H(\theta)\}$.

Figure 5.4 shows a visualization of the GC-dependent group A bacterial Shine Dalgarno RBS model for GC range [35,40) (i.e. $\theta = ([35,40), A)$). The first two rows show the merged motif models for clusters 0 and 1, respectively. In each row, the first box shows relative entropy logo of the merged motif model. This is followed by four box plots showing the probability of each letter at each position of the motif. The box widths show the

standard deviations of probabilities over the input motif models. In this case, we see that all motif models used have similar probability values per position.

The final row shows three types of information. First, it shows the clustering of the motif consensus sequences, with the numbers indicating the number of motifs that have a given consensus. Following that are the probabilities of the cluster prior model; we see that 60% of motifs are assigned to cluster 1, and 40% to cluster 0. Finally, we see the merged spacer models per cluster.

Figure 5.5 shows a similar visualization of the promoter model for $\theta = ([60, 65), C)$.

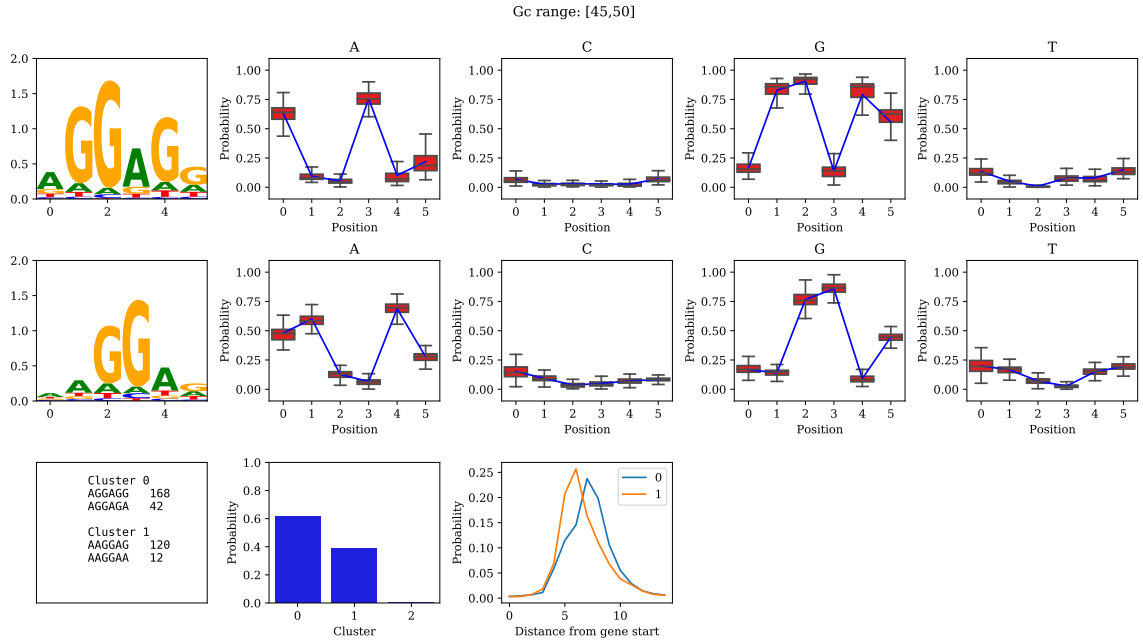


Figure 5.4: An visualization of the SD-RBS GC-dependent model constructed in the GC content range of [45, 50]. The top two rows show the merged motif models for clusters $h = 0$ and $h = 1$, respectively. In these two rows, the first column shows the motif logo computed by relative entropy, followed by the positional probability values for each letter the motif. In the bottom row, from left to right, we have: the clustered consensus sequences, the prior probability of each cluster, and the average position distributions of each cluster.

5.3.5.3 Prediction Step: Finding a motif in a non-coding sequence

Consider an upstream DNA sequence U of length L (typically, L is 20 (bacteria) or 40 (archaea) nucleotides). We can find the best position ϕ and score V of the motif in that

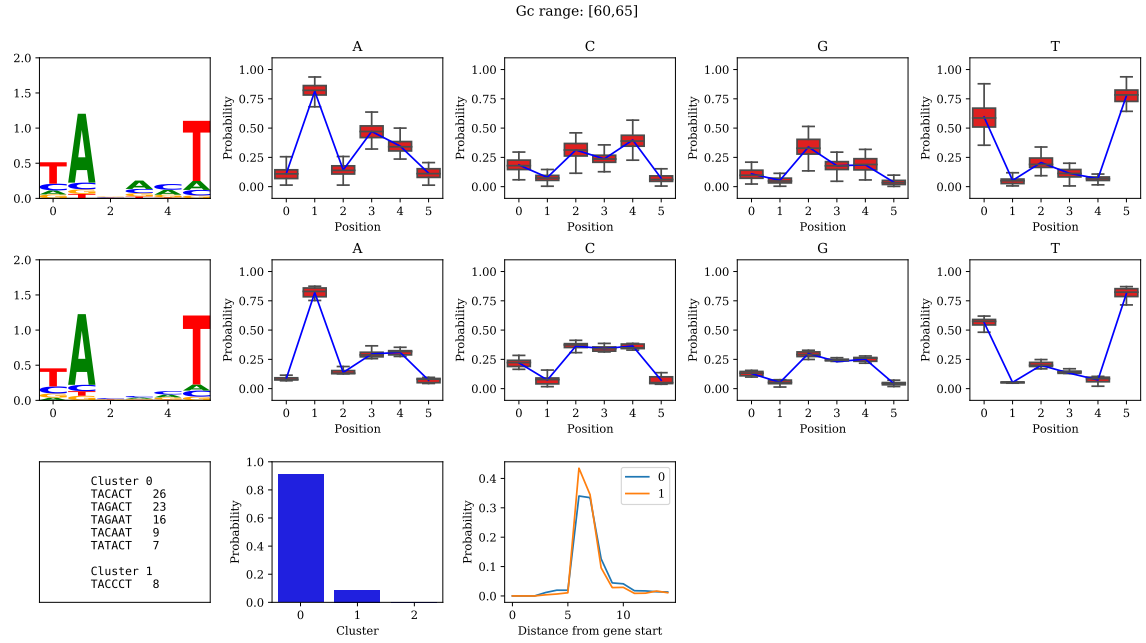


Figure 5.5: An visualization of the bacterial promoter GC-dependent model constructed in the GC content range of [60, 65]. The description of the graph components is similar to that of Figure 5.4.

sequence by maximizing over all clusters.

$$\phi = \underset{0 \le i < L-w}{\operatorname{argmax}} \max_{h} \left[p_H(h) \times S_g(L-w-i, h) \times \prod_{j=1}^{w} M(j+h, U_{i+j}) \right] \tag{5.5}$$

$$V = \max_{h} S_g(L-w-\phi, h) \times \prod_{j=1}^{w} M(j+h, U_{\phi+j}) \tag{5.6}$$

Note that while the best position uses all components of the model to determine the highest scoring motif in U, the score V that is reported back, to be used in the Viterbi algorithm, does not use the cluster prior probability $p_H(h)$. This is done to maintain a similar motif score magnitude between the motifs and other components (such as coding regions and start-context models) that are used in the HMM in GeneMarkS-2. It also makes it simpler to compare RBS motif scores to promoter motif scores when the number of clusters is different between them. Note that for groups with both promoter and RBS

model, the model with the higher score is selected for each gene.

5.3.6 MetaGeneMarkS Pipeline

Given a contig sequence, MetaGeneMarkS first detects its genetic code (4 or 11), the details of which are differed to Appendix C.3. Then, it computes the log-odds score of the sequence for each group G and selects the prediction with the highest score. Specifically, it computes

$$\underset{G\in\{A,B,C,D\}}{\operatorname{argmax}} \ \text{Log-odds}(\text{seq}|G) \tag{5.7}$$

where Log-odds is the log-odds score (by Viterbi decoding) of the sequence using MetaGeneMarkS models derived from group G genomes.

Within each computation of log-odds scores, we computes scores for each ORF using the following method. We select the set of parameters $\text{MGM}_{\theta=(g,G)}$ to compute the log-odds score for that ORF given its GC content. Note that the GC bin g is determined for each individual ORF, and not from the entire contig.

5.4 Metrics

All tools are assessed by their gene and gene-start performance. A gene-level metric measures how well a tool does at finding genes irrespective of whether the gene-start is correctly predicted, and its ability to correctly label intergenic regions as non-genes. A gene-start metric checks if the predicted gene has a correctly predicted start.

Given a set of predicted genes P and a base set B on which to compare, I use the following metrics:

- Sensitivity (Sn): A gene-level metric. Measures the number of predicted genes found in the reference set, divided by the number of genes in the reference set.
- Specificity (Sp): A gene-level metric. Measures the number of predicted genes found in the reference set, divided by the total number of predictions made.

- Gene-start Error Rate (Err): A gene-start metric. Measures the fraction of predicted genes (found in the reference set) whose start does not match the reference.

Formally,

$$\mathrm{Sn}(P, B) = \frac{M_g(P, B)}{|B|} \tag{5.8}$$

$$\mathrm{Sp}(P, B) = \frac{M_g(P, B)}{|P|} \tag{5.9}$$

$$\mathrm{Err}(P, B) = 1 - \frac{M_s(P, B)}{M_g(P, B)} \tag{5.10}$$

where $M_s(P, B)$ and $M_g(P, B)$ are the number of genes in P that match B gene-starts and gene-stops, respectively.

5.5 Challenges in performance measuring in metagenomes

We face two types of challenges when predicting genes in metagenomes. First, metagenome prediction frequently requires prediction on partially sequenced DNA fragments. This means that less genomic data may be available for a given species, and genes themselves may be partially missing. Second, the available benchmark data sets present their own share of difficulties. The limited set of verified gene-starts cannot be used as a representative test set, and the available reference sets of genes (such as RefSeq annotation) are likely to include false genes leave out true genes.

To account for fragmented DNA, I artificially construct DNA fragments by arbitrarily splitting complete genomes into individual, short sequences. This allows me to map, for instance, the verified gene-starts from the complete genome to the corresponding short fragments.

For gene-start prediction, Chapter 4 introduced StartLink+, which combines independent evidence to construct a highly reliable set of gene-starts for large numbers of genomes. In this chapter, I use a similar approach, StartLinkPH, defined as the combination of

StartLink predictions with the "Protein Homology"-based RefSeq annotation. The limitation to Protein Homology is done to remove RefSeq annotations that were a result of GeneMarkS-2, since MetaGeneMarkS models are built from GeneMarkS-2 models. As a result, I benchmark metagenomic gene-start predictions on: (1) the set of experimentally verified gene-, and (2) the set of predictions generated by the StartLinkPH on a larger set of genomes.

5.6 Results

5.6.1 Prediction on complete genomes

The first set of experiments involves finding genes in fully-sequenced genomes. This shows the performance of the metagenome predictors in a setting where all the genome's information is available to it.

5.6.1.1 Gene-Start Accuracy

Table 5.1 shows the performance on the set of experimentally verified gene-starts. In terms of metagenome algorithms, MetaGeneMarkS (177 errors) outperforms MetaProdigal (247 errors) and MetaGeneMark by a large margin (477 errors). With regards to MetaProdigal, MetaGeneMarkS does better on 6 out of the 8 genomes. Note, however, that MetaProdigal includes the Prodigal models for these two genomes, *E. coli* and *A. pernix*, in its set of 50 pre-trained models. Furthermore, *E. coli*'s genes were used in fine tuning Prodigal's gene-start model [36].

Interestingly, MetaGeneMarkS also outperforms Prodigal (196 errors) despite the latter training native models for each of the genomes, and gets close to GeneMarkS-2 (147 errors) on which its models are based. Note that all tools made a gene-start prediction for more than 99% of these genes.

Figure 5.6 shows the gene-start error rate of predictions for 438 genomes with StartLinkPH predictions, as well as the gene-level sensitivity on this reference set. MetaGeneMarkS has

Table 5.1: The number of errors in gene-start prediction on the set of experimentally verified gene-starts. The table compares metagenome algorithms (MetaGeneMark, FragGeneScan, MetaGeneAnnotator, MetaProdigal, and MetaGeneMarkS) as well as algorithms designed to run on single, complete genomes (GeneMarkS-2 and Prodigal). Highlighted are the lowest (black, **bold**) and second lowest (red, bold) values per row. The tools were executed on the complete genome sequences.

Tool	**MGM**	**FGS**	**MGA**	**MetaProdigal**	**MetaGeneMarkS**	**Prodigal**	**GMS2**
A. pernix	40	62	16	**3**	19	**3**	6
D. deserti	45	78	43	56	**20**	48	**14**
E. coli	129	101	34	**19**	34	**18**	28
H. salinarum	31	118	45	44	**10**	15	**7**
M. tuberculosis	134	117	97	86	**67**	77	**65**
N. pharaonis	13	52	19	6	**4**	5	**3**
R. denitrificans	74	50	43	25	**17**	26	**19**
S. PCC	11	4	12	8	7	**4**	**5**
Total	477	582	309	247	**178**	196	**147**

the lowest average error rate per genome (0.06), followed by MetaProdigal (0.08), MGA (0.11), MetaGeneMark (0.13), and finally FGS (0.14). Interestingly, MetaGeneMarkS has a lower error rate than MetaProdigal in low- and high-GC genomes, and an almost equal error rate in the mid-GC range. All tools have a high sensitivity rate, indicating that they find more than 99% of genes irrespective of gene-start, except FGS, which goes down to 97%.

5.6.1.2 Gene-Level Accuracy

The accuracy at the level of genes typically measures (1) false positive predictions, i.e. intergenic regions predicting as genes, and (2) false negative predictions, i.e., missed genes. In both, we count a gene as "found" judging only by whether its stop codon was detected; i.e. gene starts are ignored. Both these metrics introduce challenges.

- False Positives: Suppose a region is predicted as a gene, but that it is not present in the reference annotation. Given that annotations are often constructed by computational means, it is possible that this prediction is an actual gene missed by the annotation.

- False Negatives: Similarly, an annotated gene may be a false positive in the annota-

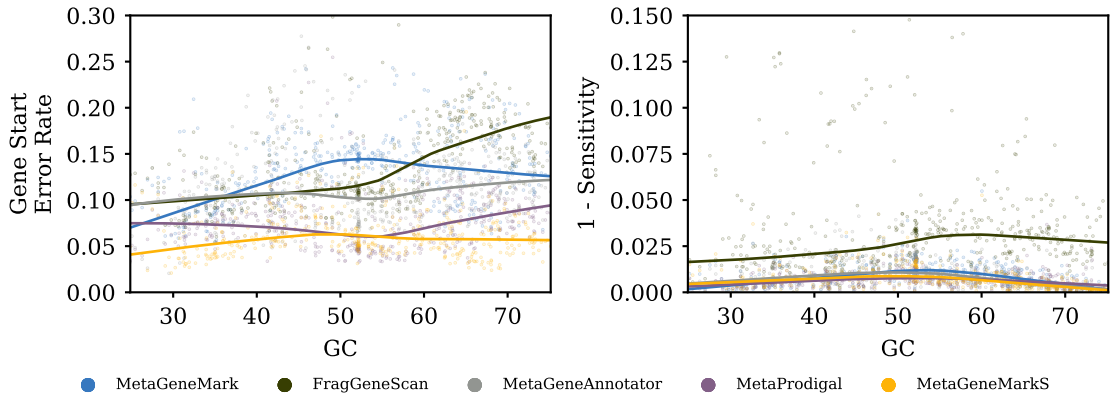


Figure 5.6: The gene-start error rate and gene-level sensitivity as a function of genome GC content. This is computed over the StartLinkPH predictions for 438 genomes.

tion itself, and therefore correctly predicted as intergenic by the algorithm.

Nevertheless, the comparisons below will use RefSeq annotation as a reference point, and very small fluctuations can be ignored.

Table 5.2 shows the number of missed genes when comparing against RefSeq annotation of the 8 genomes. . The annotated genes are split into bins based on their length. From the set of metagenome predictors, MetaProdigal misses the fewest genes, followed by MetaGeneMarkS and then MGA. That said, the difference between these three tools is less than 0.3%.

Table 5.3 shows the number of predicted genes that are not present in the RefSeq annotation. MetaGeneMarkS has the lowest number of such predictions, followed by MetaProdigal which makes an additional 230 predictions. Note that the numbers are similar to what Prodigal and GeneMarkS-2 predict.

Similarly, we see consistently high sensitivity and specificity rates across GC for all tools, except FGS whose specificity drops in the mid-GC range to 0.9 (Figure 5.7).

5.6.2 Prediction on genome fragments

One way to simulate metagenomic prediction is to split a complete genome into smaller fragments, and run prediction independently on each fragment. For a given fragment size

Table 5.2: The number of missed genes from all RefSeq-annotated genes in the set of genomes shown in Table 5.1. Genes are split in bins based on their length (as determined by the annotation). Bold values indicate the minimum values per bin across the metagenome algorithms (Prodigal and GMS2 are included in the table as reference points to how well native models can do).

Bins	**<150**	**150-300**	**300-600**	**600-900**	**>900**	**Total**	
Annotated	224	2,286	6,360	5,981	11,492	26,343	
	Missed annotated genes (FN)					Number	Percent
MetaGeneMark	128	315	301	51	34	829	3.15
FragGeneScan	159	448	608	298	236	1,749	6.64
MetaGeneAnnotator	167	**260**	248	33	26	734	2.79
MetaProdigal	103	291	**237**	**30**	14	675	2.56
MetaGeneMarkS	**96**	273	244	34	**11**	**658**	**2.50**
Prodigal	87	262	230	21	15	615	2.33
GeneMarkS-2	91	276	264	25	8	664	2.52

Table 5.3: The number of predicted genes not found in the RefSeq annotation. Genes are split in bins based on their length (as determined by the annotation). Bold values indicate the minimum values per bin across the metagenome algorithms (again, Prodigal and GMS2 are included in the table as reference points to how well native models can do).

Bins	**<150**	**150-300**	**300-600**	**600-900**	**>900**	**Total**
Annotated	224	2,286	6,360	5,981	11,492	26,343
	Predictions not in annotation (FP)					Number
MetaGeneMark	257	714	**291**	80	3	1,345
FragGeneScan	354	1,601	575	100	218	2,848
MetaGeneAnnotator	**32**	923	441	52	**0**	1,448
MetaProdigal	276	715	311	**42**	70	1,414
MetaGeneMarkS	244	**700**	301	86	32	**1,363**
Prodigal	310	722	256	71	77	1,436
GeneMarkS-2	285	601	248	84	17	1,235

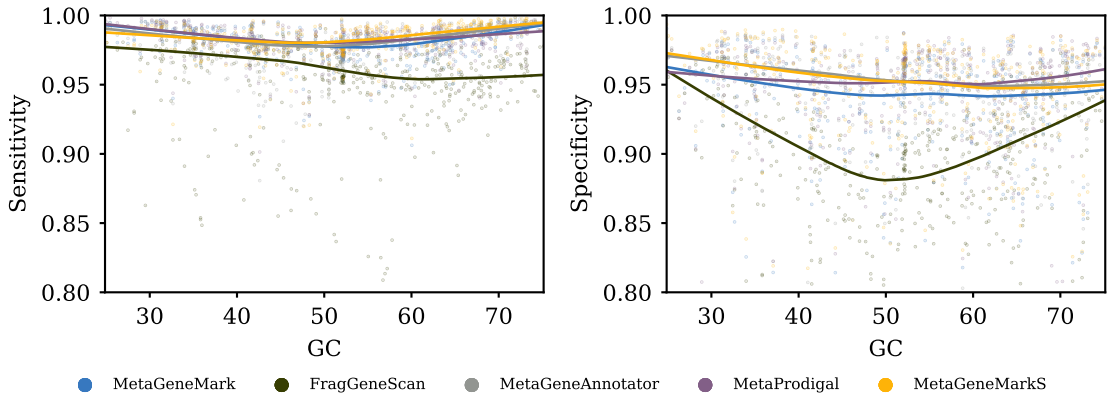


Figure 5.7: The sensitivity and specificity across GC, computed over StartLink+ predictions for complete genomes.

f, the genome is first split into contiguous pieces each of size f nucleotides. Each tool is then used to predict genes in the individual fragments. To measure accuracy on a given reference set, we map that set of genes to the split fragments; i.e. if a gene maps to two split fragments, then that annotated gene label is also split across the fragments.

5.6.2.1 Gene-start Accuracy

Measuring gene-start performance on genomic fragments can result in partial genes, i.e. cases where only part of the gene is included in the fragment. When the true start of a gene is not in the fragment, then a correct prediction would be to indicate that this is a partial gene at the gene start. The results in this section are computed independently for complete and incomplete genes.

Figure 5.8 shows the gene-start error on the set of genes with verified starts when executed on split fragments. The size f is varied from 1K to 5K nucleotides. The relative performance between MetaGeneMarkS, MetaProdigal, and MGA is similar to their performance on the complete genomes (Table 5.1).

For partial genes whose true start is not present in the fragment, MetaGeneMarkS and MetaProdigal have the lower error rates near zero, and, along with MetaGeneMark, find the largest number of such genes. Note that the larger error rate of MGA, MetaGeneMark,

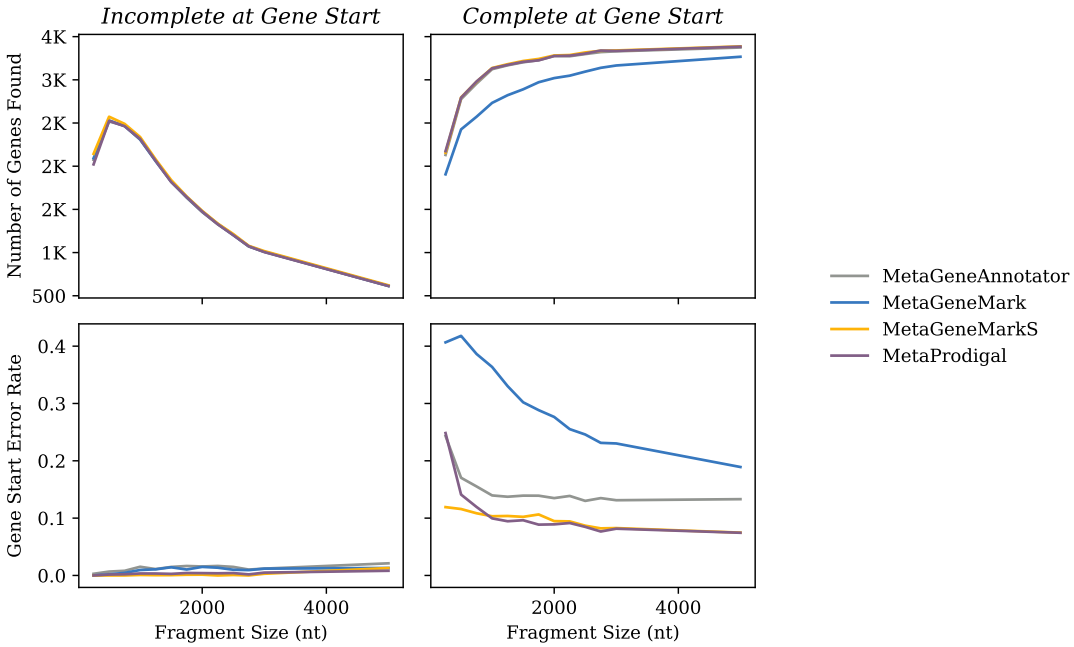


Figure 5.8: The gene-start error of MetaGeneMark, MetaGeneMarkS, and MetaProdigal on the set of verified gene-starts. The genome sequences are split into shorter fragments of size f, where f is varied from 1K to 5K nucleotides. The error rates are computed separately for genes incomplete at the gene-start and the rest. Note: For the top-right graph, MetaGeneMarkS's performance is hidden below MetaProdigal's and MetaGeneMark's curves.

and FGS indicates that they predict shorter gene fragments instead of labeling them as incomplete. They also find fewer such genes. The ranking is similar on the set of genes whose start is within the fragment. However, for very short chunk sizes, MetaProdigal's error rate rises quickly despite there being more than 2,000 genes in that range.

Figure 5.9 shows the gene-start error rate and the number of reference genes found across GC, for the set of genomes with StartLinkPH predictions. The results are shown for different fragment sizes, 250, 500, 1000, and 1500 *nt*. We can see that MetaGeneMarkS outperforms MetaProdigal on short fragments of length $<1000nt$. For larger fragment sizes, the behavior is similar to that we saw on complete genomes.

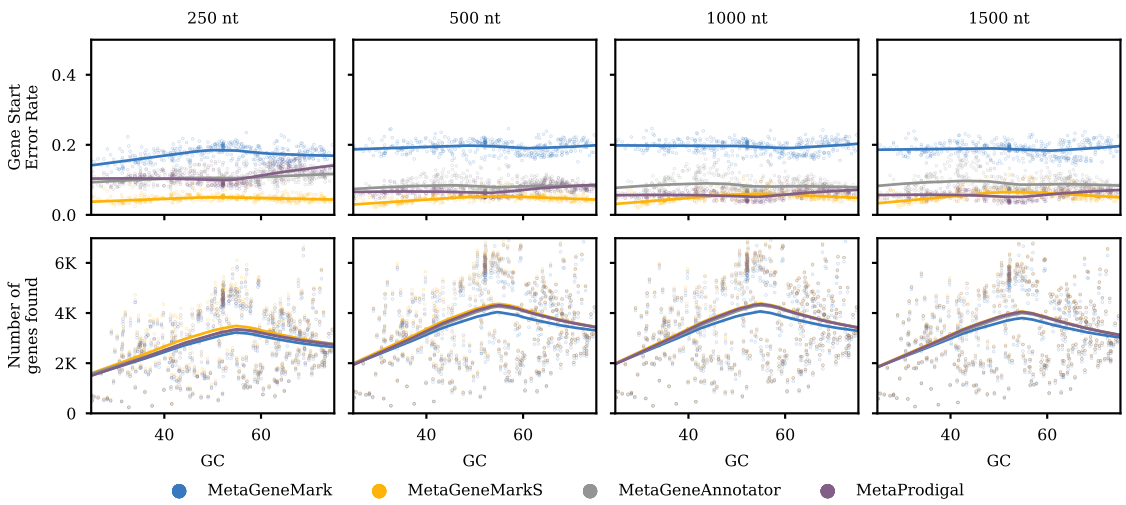


Figure 5.9: The gene-start error on the set of StartLinkPH gene-starts. The genome sequences are split into shorter fragments of size f, where f is varied from 1K to 5K nucleotides.

5.6.2.2 Gene-level Accuracy

Figure 5.10 shows the sensitivity and specificity as a function of the fragment size. For MetaGeneMark, MetaGeneMarkS, and MetaProdigal, the sensitivity rate is consistently high and the tools recover most annotated genes. Interestingly, MGA has a very low sensitivity rate when fragments are short; in particular, for fragments less than 1000 *nt* long, MGA misses more than half the number of annotated genes. On the flip side, it is the only tool that maintains a low number of false positive predictions for short fragments. Moreover all tools apart from FGS have similar specificity rates. This means that for short fragments, MGA prefers lower false positive rates despite finding much fewer genes, while the remaining tools prefer finding most genes, even if it means over-predicting on short fragments. As the size grows beyond 2000 *nt*, the number of false positive predictions across all tools stabilizes near zero, while maintaining high sensitivity and specificity rates.

We see a similar pattern when comparing to StartLinkPH prediction on short fragments. The sensitivity and specificity across GC is similar to what was shown in complete genomes, with an average score of 0.3% per each metric for MetaGeneMarkS, MetaProdi-

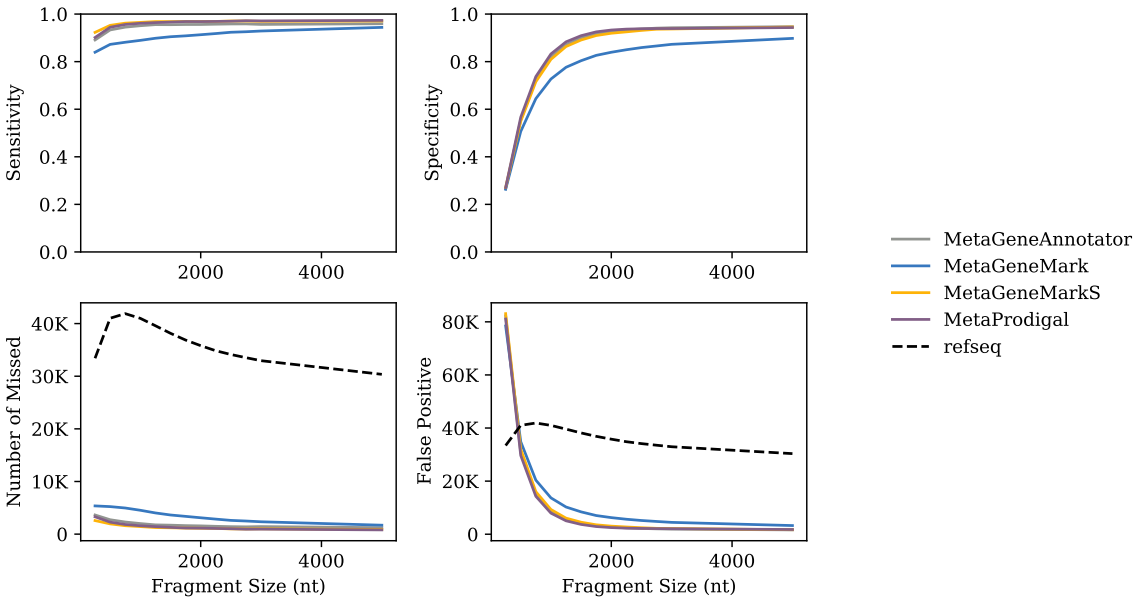


Figure 5.10: The gene-level sensitivity and specificity rates on the set of 8 verified genomes, as a function of the genome split size f. The dotted lines in the bottom plots shows the total number of RefSeq annotated genes,

gal, and MGA.

5.7 Discussion

5.7.1 Visualizing motif similarities and differences

In the above, I described how GC content, GeneMarkS-2 group, and genome type, i.e. archaea or bacteria, are used to separate genomes into groups from which independent models can be learnt. I also showed how these features can discriminate between these groups for start-codon and start-context probabilities. In this section, I show a similar analysis for RBS and promoter motifs.

In GeneMarkS-2, motif models are represented as a positional Markov model. The primary difficulty with mapping motif models across GC is that positions within this Markov model are not consistent across genomes. For example, suppose we build two motif models form two separate genomes, and that their consensus sequences have the form GGAGGC and AGGAGG . Clearly, the probabilities of the first position in the first motif are much

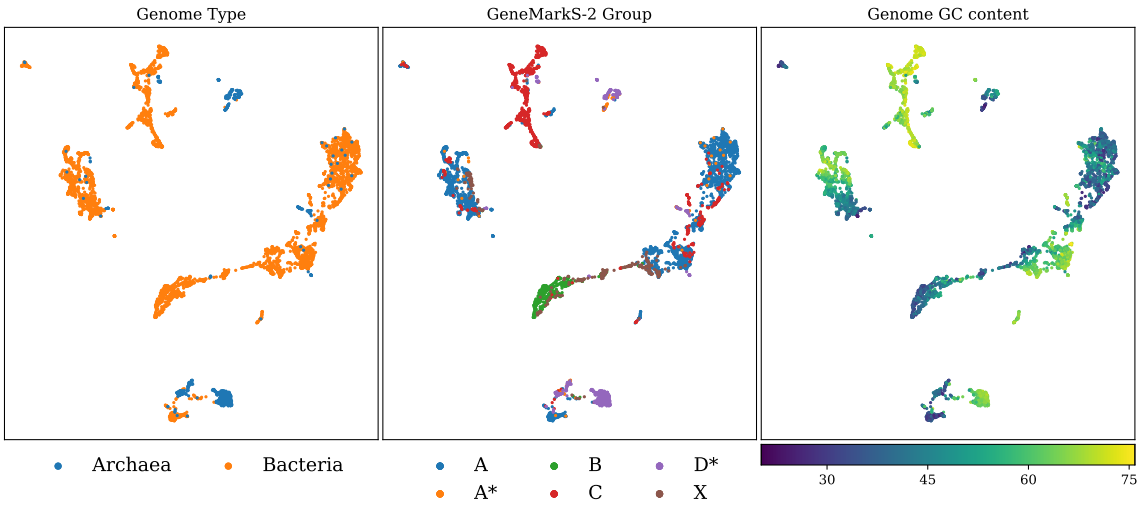


Figure 5.11: A visualization of the relationship between RBS models for 800 archaea and 5200 representative bacterial genomes. The 4x6 RBS positional Markov models are derived by running GeneMarkS-2 on each genome, and are then transformed to this 2D space using UMAP. The transformation is then colored based on three criteria: (1) if the RBS comes from an archaea or bacteria genome, (2) what GeneMarkS-2 group was assigned to the genome, and (3) GC of the genome. Note: the "*" after a group indicates that this is from an archaeal genome.

more likely to be related to those in the *second* position of the second motif.

This type of shift is very common in image classification applications. Two images can show the same object but shifted by a few pixels left or right, or even rotated. Complicating the matter further is the high dimensionality of the data. In order to visualize the similarity between these images, researchers use dimensionality reduction techniques such as UMAP and tSNE that map high-dimensional data into a 2D space, preserving a non-linear similarity measure between the datapoints. In this analysis, I use UMAP to visualize the relationships between RBS models, mapping them from a 24D space into a 2D space.

Figure 5.11 shows the visualization of RBS positional Markov models using UMAP. After the transformation, the datapoints are colored based on multiple criteria such as genome type, i.e. archaea or bacteria, GeneMarkS-2 group, and GC content.

The models built from archaea genomes are located separately from those of bacteria genomes, though note that the number of archaea genomes is rather small, which leads

to a dominance of bacterial genomes in that figure. More interestingly, we see a strong separation between genomes from different GeneMarkS-2 groups. Group B genomes are obviously separable from the rest due to their non-canonical, A-rich RBS. It is interesting to see that RBS models from Group C are separated from Group A, despite both being Shine Dalgarno based. It is unclear if this is a byproduct of the slight difference in approach to how each is constructed in GeneMarkS-2, or if it has a more biologically-relevant explanation. Finally, we see a strong dependency on GC as well. Overall this makes it clear that similar RBS models can be gathered using these features.

5.7.2 Effects of Leaderless Model

A primary advantage that MetaGeneMarkS has over other metagenome algorithms is its leaderless transcription model. The effect is clear when we look at the genomes from the verified set that have a large number of leaderless transcripts, specifically, *D. deserti*, *H. salinarum*, *M. tuberculosis*, and *N. pharaonis*.

On this set, MetaGeneMarkS makes 104 fewer gene-start errors than MetaProdigal. MetaGeneMarkS outputs whether an RBS or a promoter was used in predicting each start, and MetaProdigal outputs whether an RBS or "No RBS" was used.

Let us take the genes from this set whose starts have been predicted correctly by MetaGeneMarkS and incorrectly by MetaProdigal as compared to the verified set. Table 5.4 shows the number of genes that MetaGeneMarkS predicts correctly and the associated RBS or leaderless label. It also shows what MetaProdigal's incorrect predictions are labeled as for these genes.

For the gene-starts labeled as RBS by MetaGeneMarkS, MetaProdigal makes an error even though it uses its RBS model. For those labeled as leaderless by MetaGeneMarkS, however, MetaProdigal frequently makes an error because it tries to use an RBS model in a situation where no RBS exists. For the total of 123 genes correctly predicted as leaderless by MetaGeneMarkS, 83 of them (68%) are labeled as RBS by MetaProdigal.

Table 5.4: The genes where MetaGeneMarkS makes a correct gene-start prediction, and MetaProdigal makes an incorrect prediction. Shown are the MetaGeneMarkS label for these genes (RBS or Leaderless), and whether MetaProdigal labels those genes as using RBS or not. This is done for the set of genes with verified gene-starts.

Genome	**MGM2**	**MetaProdigal**	
		RBS	**No RBS**
D. deserti	**RBS**	2	0
M. tuberculosis		12	3
H. salinaruum		27	2
D. deserti	**Leaderless**	30	14
M. tuberculosis		26	14
H. salinaruum		27	12

Table 5.5: The genes where MetaProdigal makes a correct gene-start prediction, and MetaGeneMarkS makes an incorrect prediction. The structure is similar to that described in Table 5.4.

Genome	**MGM2**	**MetaProdigal**	
		RBS	**No RBS**
D. deserti	**RBS**	3	0
M. tuberculosis		13	3
H. salinaruum		1	1
D. deserti	**Leaderless**	3	2
M. tuberculosis		11	10
H. salinaruum		3	4

The leaderless label for these genes is supported by analyzing their motifs as predicted by MetaGeneMarkS. Figure 5.12 shows the motif logos for this set of 123 genes. We see that the signals indicate that this is a promoters, with a localized spacer at 6 *nt* for bacteria and 23 *nt* for archaea.

A similar analysis can be done by looking at the starts predicted correct by MetaProdigal and incorrectly by MetaGeneMarkS, though there are fewer such genes; 54 compared to 169 in the previous case. In particular, of the 34 correctly predicted gene-starts labeled as "RBS" by MetaProdigal, half were labeled as leaderless by GeneMarkS-2, which likely caused the error in prediction.

Overall, this shows that MetaGeneMarkS's leaderless model is able to capture gene-starts missed by MetaProdigal.

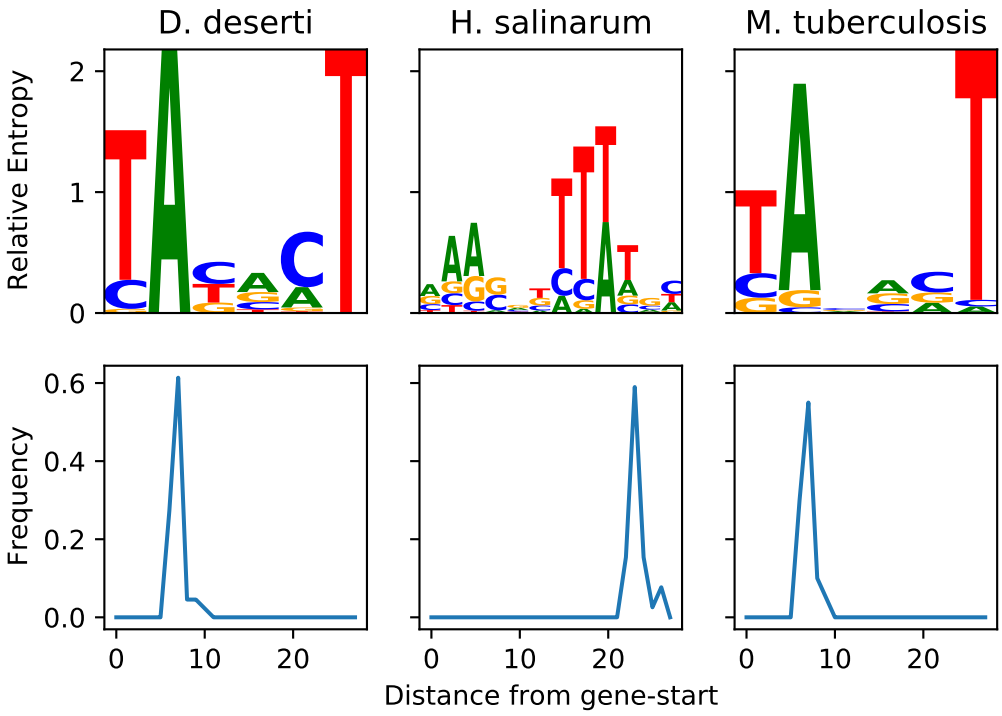


Figure 5.12: The motifs and spacer distributions constructed for the set of genes with a correct MetaGeneMarkS prediction and incorrect MetaProdigal prediction, when compared to the set of verified starts. Specifically, this is for the 44 *D. deserti*, 40 *M. tuberculosis*, and 39 *H. salinarum* genes labeled as leaderless by MetaGeneMarkS.

5.7.3 Genetic-code 4 motif models

From the set of 5,238 representative bacterial genomes, only 101 are labeled as genetic-code 4. Nevertheless, we can use this set to draw reasonable insights into the translation-initiation mechanisms used by these species.

GeneMarkS-2 classified these genomes into groups as follows: 58 into group A, 33 into group C, and 5 into each of groups B and X. It isn't clear by looking at the generated motifs whether the group B assignment is actually correct; i.e. whether the non-canonical RBS model is present. As such, these genomes were excluded from the remainder of this analysis.

Figure 5.13 shows the distribution of group assignments across GC, for both genetic-code 4 and genetic-code 11 genomes. This shows that leaderless transcription is common in low-GC bacteria genomes. It is not immediately clear why the density of group C genomes is low in genetic code 11 bacteria. This may be due to an over-representation of high-GC group C genomes in the set of representative genomes (and/or in the full RefSeq database itself).

Furthermore, the percentage of leaderless transcripts in these 33 genetic-code 4 genomes is similar found in genetic-code 11 genomes, shown in [58]. Figure 5.14 shows distribution of leaderless transcripts in the 33 group C genetic-code 4 genomes as determined by GeneMarkS-2. The percentages range from 30 to 60, with an average of 46%.

A manual inspection of the promoter and RBS motifs found for these genomes shows strong, well-localized signals. Figure 5.15 shows the promoter and RBS motif models for four *Mycoplasma* genomes. The structure follows the typical group C, genetic code 11 genomes shown in the work presenting GeneMarkS-2 [58].

These results show clear evidence of frequent leaderless transcription in genetic-code 4 genomes. As such, motif models for group A and group C genomes were included in MetaGeneMarkS's genetic-code 4 model.

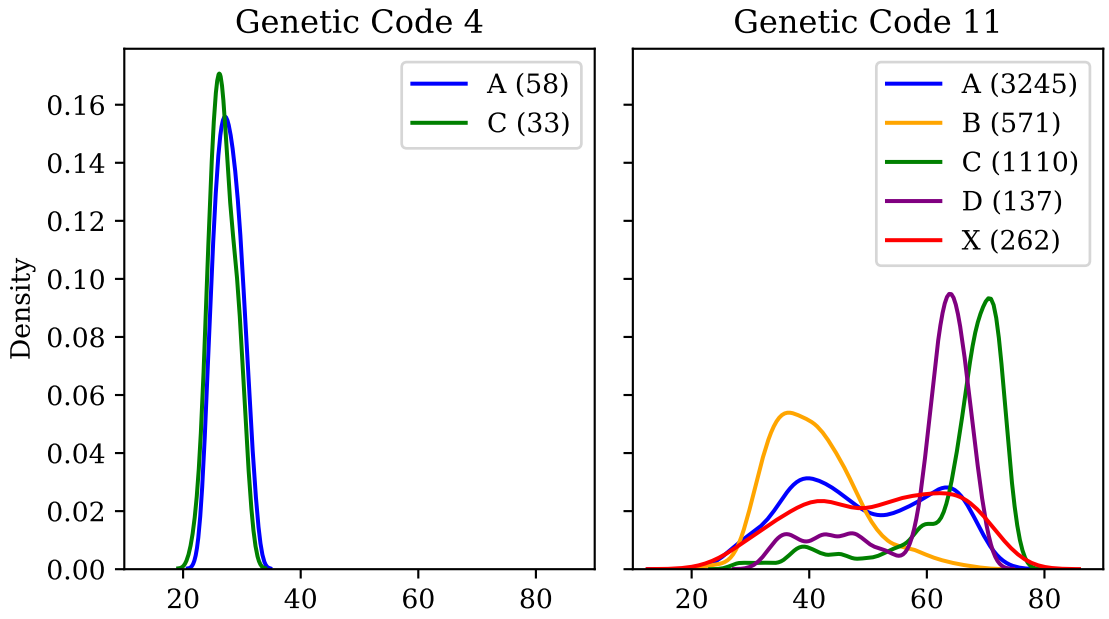


Figure 5.13: The per-group density distributions of representative bacterial genomes across GC, for genetic code 4 (left) and 11 (right). The number of genomes per group is shown in each figure's legend.

5.8 Conclusion

In this chapter, I showed how GeneMarkS-2's gene-start models can be ported into the metagenomic case. I presented MetaGeneMarkS, a metagenomic gene-finder that achieves high gene-start accuracy. It does so by explicitly modeling multiple modes of translation-initiation mechanisms, which include canonical and non-canonical RBS, and leaderless transcription in both archaeal and bacterial species.

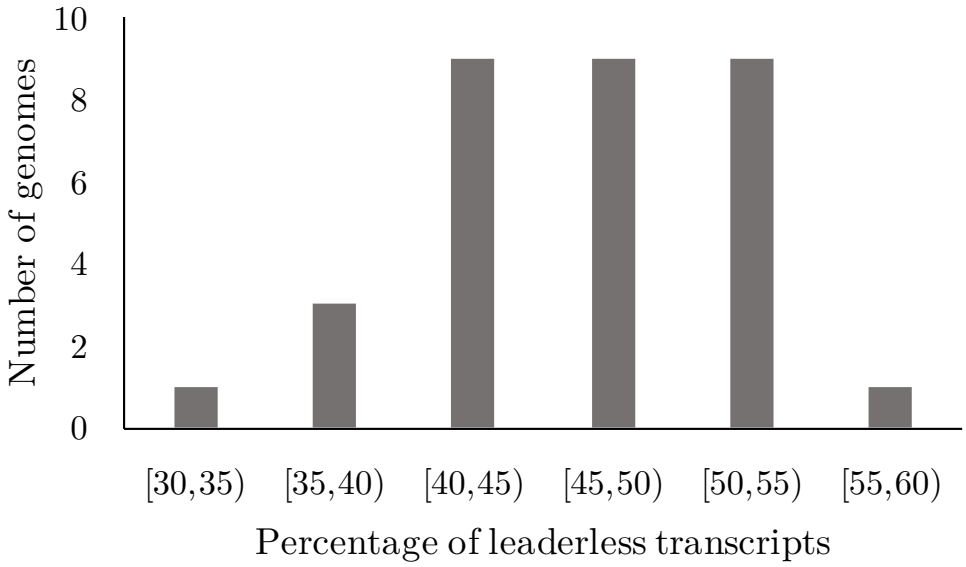


Figure 5.14: The percentage of transcripts labeled as leaderless by GeneMarkS-2, in the 33 group C genetic-code 4 genomes.

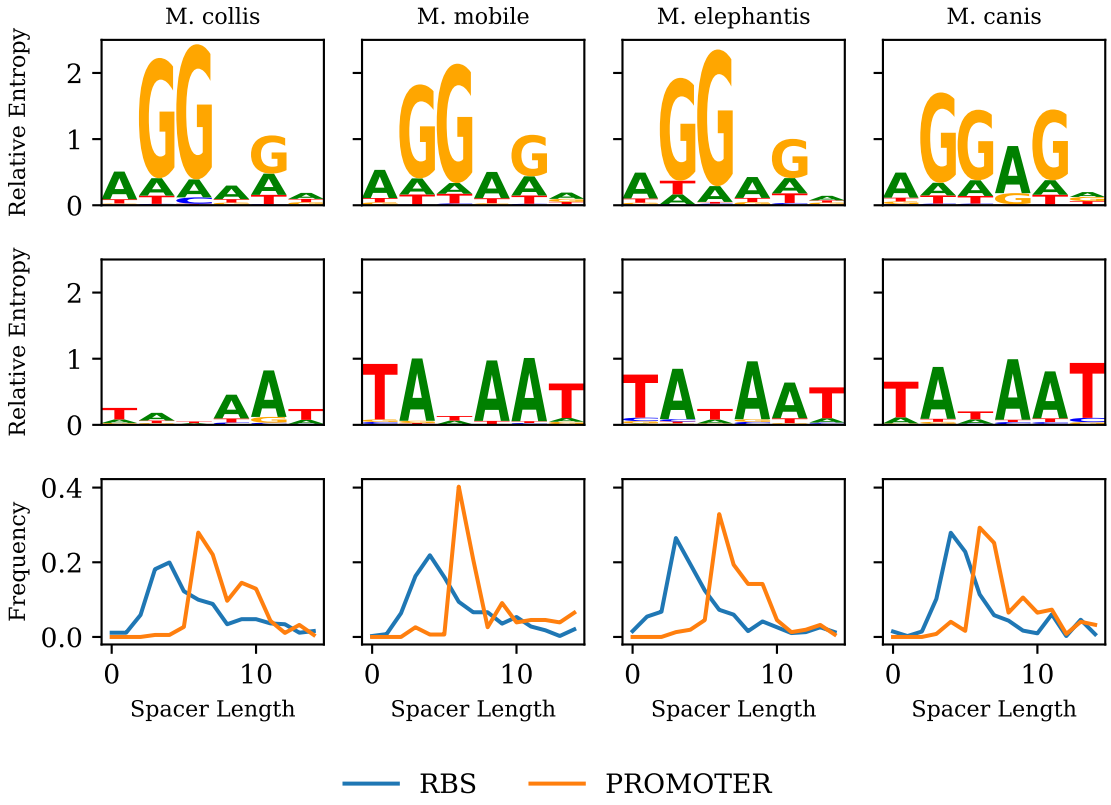


Figure 5.15: The promoter and RBS models of four *Mycoplasma*, genetic code 4 genomes. The motif logos were constructed using the relative entropy of the motif model versus the genome's GC content (representative by a zero order, uniform Markov model).

CHAPTER 6
CONCLUSION

In this dissertation, I explored the problem of prokaryotic gene-start prediction. I presented algorithmic improvements, updated our understanding of translation-initiation mechanisms in prokaryotes, and tackled the problem of a lack of proper benchmarking mechanisms.

First, I presented GeneMarkS-2, an unsupervised learning approach to gene-start prediction in complete genomes. I showed that leaderless transcription and non-canonical RBS play a big role in translation-initiation in prokaryotic genomes, and that modeling these features leads to improvements in gene-start prediction. I showed that the usage frequencies of translation-initiation mechanisms such as leaderless transcription, Shine-Dalgarno RBS, and non-canonical RBS tend to cluster together on the taxonomic tree. For example, leaderless transcription is very common in *Actinobacteria*, while non-canonical RBS are commonly found in *FCB group*. This work has been integrated into the National Center for Biotechnology (NCBI)'s pipeline for prokaryotic gene prediction (PGAP)[1]. It has since been used as part of the re-annotation of the entire RefSeq database, of more than 280,000 prokaryotic genomes, and for the annotation of newly submitted genomes. NCBI's RefSeq database has among the largest set of curated, non-redundant collection of annotated genomes, proteins, and transcripts, and is used by scientists all over the world.

Second, I addressed the lack of experimentally verified gene-starts, and how it affects our ability to accurately measure the performance of gene-start predictors. Specifically, I showed that while existing algorithms perform well on the available verified data, this performance does not generalize to the set of unverified gene-starts, which is several hundred thousand times larger. In light of this, I presented StartLink+, an approach that combines independent algorithms to filter low-certainty predictions, leaving behind a more reliable

[1]Link to PGAP description: https://www.ncbi.nlm.nih.gov/genome/annotation_prok/process/

set of predictions. This includes a new comparative genomics approach called StartLink combined with GeneMarkS-2 to produce StartLink+. This work showed that combining independent algorithms can filter most incorrect predictions, yielding an error rate of less than 2% for the remaining predictions. Finally, I showed that RefSeq annotation can differ by up to 15% with StartLink+ predictions, which suggests that there is a lot of room for improvement.

Finally, I presented MetaGeneMarkS, an approach to gene-start prediction in metagenomes. This involves finding genes in short DNA fragments, the length of which hinders GeneMarkS-2's ability to perform well. I port the models of translation-initiation mechanisms developed for GeneMarkS-2 to metagenomes by building GC-dependent models from thousands of genomes. MetaGeneMarkS outperforms its competitors at gene-start prediction. Its ability to predict gene-starts via leaderless transcription gives it a leg up compared over the competition.

I believe that this work has presented a clearer picture of the status of gene-start prediction. The main driver has been the belief that the existing algorithms' performance claims on the limited set of verified gene-starts do not adequately portray the accuracy of gene-start prediction. This still holds, and future work must continue to explore the behavior of algorithms across the increasingly diverse set of genomes, and extend the currently limited benchmarking tools and data sets that we have available.

Appendices

APPENDIX A

GENEMARKS-2

A.1 Hidden (semi) Markov models (HSMM)

Hidden Markov Models (HMMs) are a big part of our ab initio gene-prediction model. This description serves as a refresher of HMMs, and an explanation of two lesser-known concepts: an extension called Hidden Semi-Markov Models and a training approach called Viterbi Training (not to be confused with Viterbi Decoding). This part is pertinent to the description of GeneMarkS-2, and so the reader might prefer to come back to it later.

Generally, a hidden Markov model (HMMs) is a common probabilistic graphical model used in situations where sequential dependencies in the data exist (e.g. time series problems, speech and text analysis). With genes being a contiguous fragment of DNA with chemical and physical properties holding nearby nucleotides together, it is easy to see why HMMs are a reasonable candidate for gene prediction.

A naive HMM design for gene prediction can model two "states" of DNA: coding (i.e. gene) and non-coding (i.e. DNA fragment not containing a gene). Treating HMMs as a generative story for DNA modelling, we can think of an HMM generator spewing out DNA by first generating a non-coding region, followed by a coding region, then a non-coding, then a coding, etc, up until we get a full DNA sequence. This process is also referred to as sampling from the probability distribution over all DNA configurations defined by the HMM.

The first limitation of this model comes from the fact that a standard HMM, by its mathematical definition, has the property that the length of a consecutive fragment generated by any of its states must follow a geometric distribution. In other words, following the above story would mean that the lengths of the gene fragments follow some geometric distribu-

tion. However, it has been shown that this assumption does not hold for any given genome; and that a gamma distribution for gene lengths is much more suitable [72]. This can be remedied by an extension called hidden semi-Markov model (HSMM), which allows for any length distribution to be set.

The second limitation is that this model is obviously naïve, and that it fails to capture all other properties of the DNA (such as RBS motifs, promoters, etc) that can aid in gene-prediction. I discuss a more complex HSMM representation of DNA later on. It is worth spending a bit more time talking about the general problems that HSMM can solve, especially with regards to gene prediction.

A.1.1 Decoding: Identifying gene-fragments in DNA

An HSMM representation of DNA can be used to segment (or decode) a DNA sequence into its components, such as genes, motif signals, non-coding regions, etc. This is done by simply finding the segmentation s (from all possible segmentation) that maximizes the expression $P(s|DNA)$.

While this might seem daunting, an already-established, dynamic-programming approach can find this segmentation in $O(T * |S|^2)$, where T is the length of the sequence and S is the predefined set of states that characterize any DNA sequence, such as coding and non-coding regions, RBS, promoter, start and stop codons, etc.

A.1.2 Parameter Estimation: Learning an HSMM representation of DNA

In order to perform the decoding step, however, an HSMM is required. If we assume that we are given a DNA fragment with labelled segments (e.g. a fragment where all genes, motifs, etc, are identified), an HSMM can be learned through a simple maximum-likelihood approach. If λ is our model representation, then we find the λ that best fits the data by maximizing

$$P(\lambda|DNA, segmentation) \tag{A.1}$$

A.1.3 Viterbi Training: Learning an HSMM without a labelled DNA fragment

Generally, we are only provided with an unlabeled DNA fragment and asked to label it. In other words, we have no way of learning a native representation of this DNA without it being labeled, and we cannot label it without having a model representation of it. As with many cyclical problems, an iterative method provides a reasonable solution

1. Begin with a crude and basic model λ^0 (more on that later)
2. Get an initial segmentation $s^0 = \text{argmax}_s\, P(s|DNA, \lambda^0)$
3. Repeat for i=1,2,... until convergence:
   (a) Learn: $\lambda^i = \text{argmax}_\lambda\, P(\lambda|DNA, s^{i-1})$
   (b) Decode: $s^i = \text{argmax}_s\, P(s|DNA, \lambda^i)$
   (c) Output the final segmentation

Viterbi Training is similar to the more standard Expectation Maximization (EM) approach, also called Baum Welch. The main difference is that while Baum Welch estimates parameters by considering all possible segmentations, Viterbi training only uses the previously learnt segmentation (in this case, s^{i-1}). This makes it computationally faster and more memory efficient, without a large reduction in accuracy [43].

A.2 Principal equations of the Viterbi algorithm in the log-odds space

In GeneMarkS-2, the number of states in the generalized hidden Markov model (GHMM) increased significantly compared to GeneMarkS.[1] To simplify the Viterbi algorithm implementation, we move from the standard use of (log) probability values, to log-odds scores,

[1] A historical view: when I first came to this project, much had been done by Dr. Shiyuyun Tang and Dr. Alexander Lomsadze, under the advisement of Dr. Mark Borodovsky, with regards to gene-level prediction and the local GC-adaptation of the heuristic models. The gene-start models (my focus) had so far been kept as it was in the original version of GeneMarkS. Proper acknowledgement must be made of the work by my collaborators especially with the powerful improvements of GeneMarkS-2's gene-level accuracies.

i.e. where the probability of emission of a sequence fragment along a given path of the hidden states was divided by the probability of emission of the same fragment from the non-coding state. When we compared the maximum value path in the log-odds space to the maximum likelihood path in the probability space, there was little to no difference. For the models of the first and higher orders, the log-odds maximum value path was a close approximation of the maximum likelihood path. A comprehensive testing showed that the difference between the two types of implementations was concerned with ~3 genes out of 1,000. It was not clear which approach is more accurate for real data, since the range of possible errors in the test sets of validated genes was comparable with the effect we wanted to estimate.

In GeneMarkS-2, the gene prediction step in the first iteration did not use any species-specific parameters, i.e. the parameters of the native model. At this iteration the coding (M_{cod}) and non-coding (M_{non}) models for every candidate gene were taken from the array of heuristic "atypical" models. The models selected have a GC index matching the GC composition of the candidate gene. Thus selected models were used to compute the "content" (or compositional) component of the gene score Equation (A.2). The use of the GC index eliminated the computationally taxing need to visit all the 82 states corresponding to the "atypical" models.

Still, the log-odds formulation excluded some alternative paths that could be present in a full GHMM implementation. For instance, we did not directly compare the log-odds score produced by the typical model to that of the atypical model. Rather, we first selected the type of the model of protein-coding sequence (atypical or typical (native)) by comparing the probabilities of the two models, as shown in Equation (A.4). Here, $gene\_type$ was set to the type (typical or atypical) depending on which of the two coding models in Equation 2 (M_{native_coding} vs $M_{atypical\_coding}$, respectively) yielded the larger value. We then used the models of that type (i.e. typical or atypical versions of M_{cod} and M_{non}) to compute the log-odds scores defined in Equation (A.5).

For a potential protein-coding sequence $x_1x_2 \ldots x_n$ with start codon $x_1x_2x_3$, stop codon $x_{n-2}x_{n-1}x_n$, and upstream sequence $x_{-20}x_{-19} \ldots x_{-1}$, the gene start score was defined by Equation (A.6) and the rest of the protein-coding score was defined by Equation (A.5). Here, $y_1y_2 \ldots y_k$ denotes the sequence of the (potential) RBS or promoter related box, k is the site length, ϕ denotes the GC content, and Ma denotes the sequence model associated with state a. The length distributions of prokaryotic protein-coding and non-coding regions (approximated by a gamma-function and an exponential function, respectively), contributed the duration values into Equation (A.5). The last term in Equation (A.5) is the log-odds score determined by these durations. The constant C depends on parameters D_c and D_n of the gamma (protein-coding) and exponential (intergenic, non-coding) length distributions, respectively.

Finally, the overlapping genes were penalized based on the length of the overlap. In particular, for overlapping genes a and b with lengths La and Lb, respectively, and length of overlap m, a penalty S_{ovlp} was added to the score Equation (A.7).

$$S_{gene} = \log \frac{P(x_1x_2x_3|M_{start\_codon})}{P(x_1x_2x_3|M_{non})} + \log \frac{P(x_4 \ldots x_{n-3}|M_{coding})}{P(x_4 \ldots x_{n-3}|M_{non})} \tag{A.2}$$

$$+ \log \frac{P(x_{n-2}x_{n-1}x_n|M_{stop\_codon})}{P(x_{n-2}x_{n-1}x_n|M_{non})} + \log \frac{Cn^2e^{-\frac{n}{D_c}}}{e^{-\frac{n-1}{D_n}}} \tag{A.3}$$

$$gene\_type = \text{argmax}\left(P(x_{16} \ldots x_{n-3}|M_{native\_coding}), P(x_{16} \ldots x_{n-3}|M_{atypical\_coding})\right) \tag{A.4}$$

$$
\begin{aligned}
S_{CDS} = & \log \frac{P(x_{16} \ldots x_{n-3}|M_{coding})}{P(x_{16} \ldots x_{n-3}|M_{non})} \\
& + \log \frac{P(x_{n-2}x_{n-1}x_n|M_{stop\_codon})}{P(x_{n-2}x_{n-1}x_n|M_{non})} \\
& + \log \frac{Cn^2 e^{-\frac{n}{D_c}}}{e^{-\frac{n-1}{D_n}}}
\end{aligned} \tag{A.5}
$$

where M_{coding} and M_{non} is selected based on the $gene\_type$ value.

$$
\begin{aligned}
S_{start} = & \log \frac{P(x_1x_2x_3|M_{start\_codon})}{P(x_1x_2x_3|M_{non})} + \log \frac{P(x_4 \ldots x_{15}|M_{down\_signal})}{P(x_4 \ldots x_{15}|M_{non})} \\
& + \frac{P(x_0x_{-1}x_{-2}|M_{up\_signal})}{P(x_0x_{-1}x_{-2}|M_{non})} \\
& + \max \begin{cases}
\log \frac{P(y_1 \ldots y_k|M_{rbs})}{P(y_1 \ldots y_k|M_{non})} + \log \frac{P(l|M_{rbs\_spacer})}{e^{-\frac{l+k}{Dn}}} & \text{for RBS} \\
\log \frac{P(y_1 \ldots y_k|M_{promoter})}{P(y_1 \ldots y_k|M_{non})} + \log \frac{P(l|M_{promoter\_spacer})}{e^{-\frac{l+k}{Dn}}} & \text{for promoter} \\
\log \frac{P(x_{-20} \ldots x_{-3}|M_{extra\_up\_signal})}{P(x_{-20} \ldots x_{-3}|M_{non})} & \text{for upstream signature}
\end{cases}
\end{aligned} \tag{A.6}
$$

$$
S_{ovlp} = -m \log \left(1 + \frac{m}{2} \left(\frac{1}{L_a} + \frac{1}{L_b}\right)\right) \tag{A.7}
$$

A.3 Assigning a genome into a group

The sequence of steps. The identification of the type of regulatory site model goes in parallel with the genome group assignment. The candidate groups are tested in a particular order, and the genome is assigned to the first group for which the 'membership' test is successful. The process differs slightly for archaeal and bacterial genomes, where an archaea genome is tested against groups D, B, A, X (in that order), while a bacterial genome is

tested against groups C, B, A, X. If a genome's domain (i.e. archaeal or bacterial) is not known, then all groups are tested in the order D, C, B, A, and X. The steps are described below, and illustrated in Figure A.1.

Group D models. This model assumes a presence of both leadered and leaderless transcription. Therefore, both the promoter and the RBS models are to be determined. There are two ways that an archaeon can be assigned to Group D; the first method works well when the percentage of leaderless transcripts in the genome is high, and the second caters specifically to the case when leaderless transcripts are less frequent. The two methods only differ in the way the training sequences for the promoter and RBS models are selected.

In the first method, we select the 40 nt long fragments upstream to all FGIOs and run GibbsL to possibly detect a 12 nt long motif of the -26 box of the archaeal promoter [73]. On the other hand, we also run GibbsL on the 20 nt long fragments from all IGIO to find the 6 nt long RBS pattern. If the detected promoter motif is localized at a distance larger than 14 nt (with the 10% mode threshold) then this genome is assigned to Group D.

If this condition does not hold, then the method two is applied to detect a lower percentage of leaderless transcripts. In the second method, we choose a set of 20 nt long fragments located upstream to all the FGIOs, and single out those that show a local similarity to the extended Shine-Dalgarno sequence TAAGGAGGTGA (by checking for 4 consecutive nucleotide matches, with one possible U-G type substitution). This search divides the set of FGIOs into two sets, ones with the upstream fragments having the SD similarity (set X) and ones with upstream fragments having no SD similarity (set Y). We extend set X by adding the 20 nt fragments located upstream to all the IGIOs (expecting them to contain subsequences similar to SD-RBS). Then, we run GibbsL on set X to find the 6 nt long RBS pattern. In a parallel step, we look into set Y to select the 40 nt long fragments upstream to FGIOs and use GibbsL to find the 12 nt long motif. The rational for this is a search for the B Recognition Element (BRE) that in archaea may be located just upstream to the TATA box [73, 74]. Again, the localization distance is checked to be larger than 14 nt at

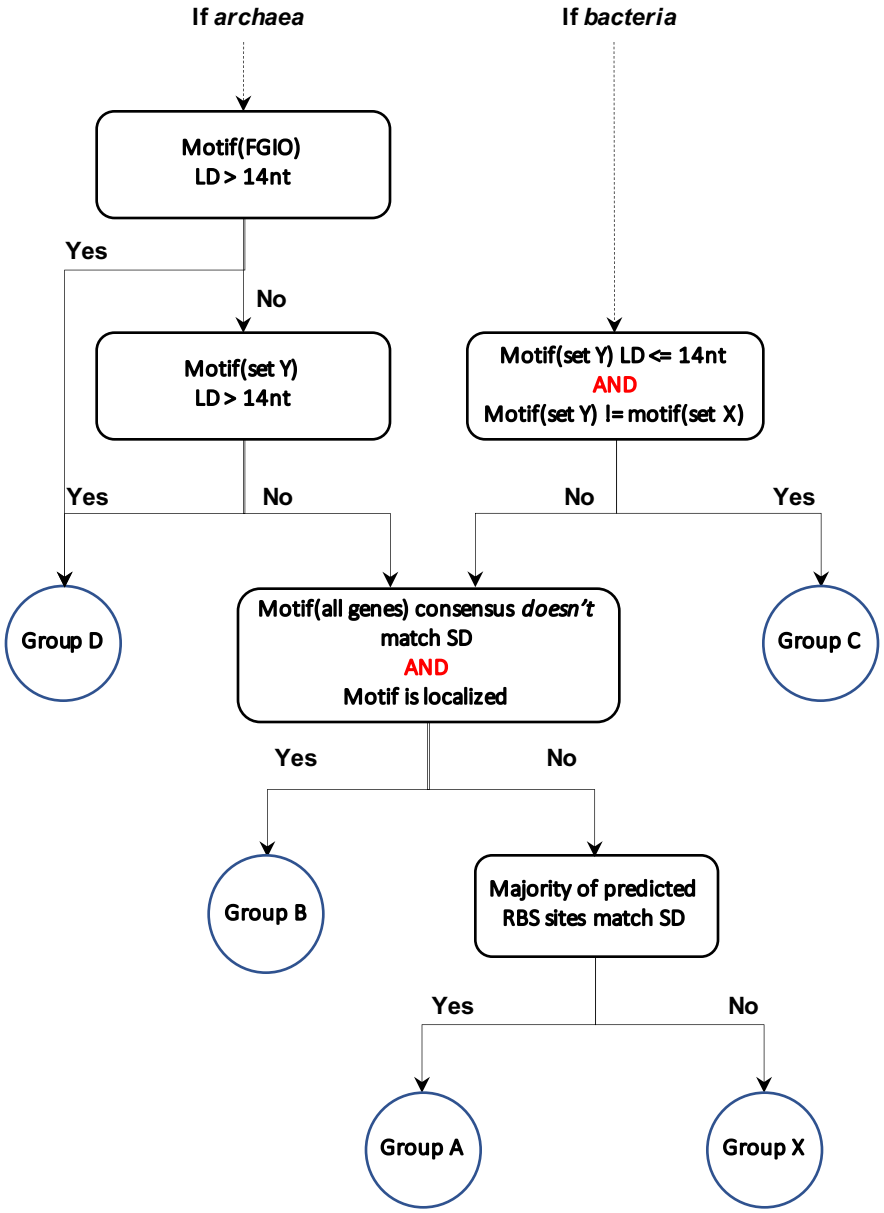


Figure A.1: This figure describes the procedure of deriving the type and parameters of the model of a sequence around gene start (models A through D, and X). Here, Motif(set X) represents the motif derived form a set of sequences X, and LD stands for the localization (peak) of the spacer distribution for that motif model.

a 10% mode threshold. If the condition is satisfied, then the genome is assigned to Group D. It is possible that set Y is sufficiently large but the training the promoter model may not have conclusive results. This may happen when the set Y contains fragments with non-SD RBS (thus, the search for similarity with the extended SD sequence would not produce the desired result). If so, we proceed to derive the Group B model (see below). An example of a Group D genome is Halobacterium salinarum where more than 70% of operons have leaderless transcription.

Group C models. This model is determined for bacterial genomes under assumption that both leadered and leaderless transcription occur along the genome. The approach is similar to the one used for the derivation of the Group D model described above. In bacteria with possible instances of leaderless transcription we model the -10 promoter box (with length 6 nt) which is the closest promoter site to TSS.

We select a set X of the FGIOs for which the 20 nt long upstream fragments have a local similarity to the extended Shine-Dalgarno sequence TAAGGAGGTGA. This set is then augmented with the 20 nt long fragments located upstream to all the IGIO genes, thus giving us the input set of fragments on which GibbsL is executed to find the 6 nt long RBS motif. Next, we take the set of 20 nt long fragments located upstream to the genes in set Y and run GibbsL to detect the 6 nt long motif of the bacterial promoter box (-10). If we find a motif with the localization distance satisfying the 25% threshold, then the genome is assigned to Group C. An example of a Group C species is *M. tuberculosis*.

Having examined many bacterial species, we observed that, in some cases, the two motifs derived from sets X and Y could be very similar. Since set Y could not produce an SD-RBS motif (given the way set Y was selected) and that set X could not produce a bacterial promoter motif (given the prevalence of IGIOs in set X), it is unlikely that the motifs from set X and Y (when similar) constitute either an SD-RBS or a promoter. Therefore, in that case, we proceed with the Group B membership test, where (all) genes may have a non-SD RBS.

Group B models. This type of model is derived for the genomes with the pattern derived for FGIOs is similar to the pattern derived for IGIOs while the consensus of this pattern differs from the one of Shine-Dalgarno. This outcome is observed, e.g., in *Flavobacteriia*, *Bacteriodia* and *Cytophagia*. Since this pattern is present in IGIOs, it cannot be related to a promoter; as such, the Group B model could be characterized as a non-SD RBS.

To identify such a case, we compare the consensus sequences of the two motifs (from sets X and Y, as described above) derived in Group C. If the two consensus sequences share three or more consecutive nucleotides (out of 6), they cannot make the distinct promoter and RBS pair as in Group C.

No test is needed for archaeal genomes since the promoter is located more than 15 nt away from gene-start. This distance is large for an RBS (thus eliminating archaea from potentially be detected as a non-SD RBS along the line of this logic).

If the matching condition is satisfied, we derive a single 6 nt long motif by the GibbsL alignment of the 20 nt upstream regions of all the genes. Next, the consensus sequence of this motif is compared to the extended Shine-Dalgarno sequence. A significant match to the extended SD sequence constitutes at least four consecutive nucleotide identities (allowing for U-G type substitutions). If such a similarity is not present, while the motif is well localized (i.e. the peak of its position distribution is more than 15%), then we conclude that the single non-SD RBS motif is in place for this Group B genome. Otherwise, we continue with the membership test for Group A (see below). An example of a Group B genome is *Bacterioides ovatus*.

Group A models. This single-motif model describes the translation initiation with the SD type RBS, the most frequent case in the prokaryotic genomes we have studied. To derive the models for this group, we run GibbsL on the set of 20 nt long upstream regions of all the predicted genes. Next, we compute the fraction of predicted RBS sites (among all predicted genes longer than 300 nt) that show a local similarity to the extended Shine-

Dalgarno sequence (see above). If such a fraction exceeds 0.5, then the genome is assigned to Group A; otherwise, we proceed with the step described below. An example of a Group A species is *E. coli*.

Group X models. Genomes that do not pass any of the above group membership tests are lumped together in Group X. It seems that these genomes mostly use leadered transcription (since no promoter can be identified near gene-start) but do not have an identifiable RBS. Still, the SD type RBS model could be valid for some genes. To derive this model, we select the genes whose 20 nt upstream regions contain a local similarity to the extended Shine-Dalgarno sequence. The common RBS motif is derived from this set by GibbsL. For all remaining genes, the algorithm derives the "extended upstream signature" model, a 2nd order positional frequency model generated from the alignment of the upstream sequences with respect to the predicted gene starts. An example of a Group X genome is *Synechosystis*.

A.4 Motif search by GibbsL

The GibbsL algorithm works with a set of N sequences $\{S^{(1)}, \ldots, S^{(N)}\}$ such as DNA sequences located upstream of the predicted gene-starts. We can assume, for simplicity, that all sequences have the same length L. Let $\boldsymbol{a} = \{a_1, a_2, \ldots, a_N\}$ be the vector of positions, where $a_n = i$ indicates the starting position of the predicted motif of fixed length W in sequence $S^{(n)}$. The part of sequence $S^{(n)}$ that does not belong to the motif is called the "background." The set of motifs, when found, are used to define the parameters of the position (non-uniform) Markov model, M_{motif}. Similarly, the backgrounds of the sequences $\{S^{(1)}, \ldots, S^{(N)}\}$ are used to defined the uniform Markov model, M_{bgd}.

The probability of the motif sequence under a zero-order Markov model is defined as

$$P(S_{i\ldots i+W}|M_{motif}) = \prod_{z=i}^{i+W-1} P\left(S_z|M_{motif}, z\right) \tag{A.8}$$

Here, $S_{i...j}$ are the nucleotides in S at positions $z = i, i+1, \ldots, j$.

Finally, we account for the positions of motifs relative to a fixed pivot. This is in contrast to other motif finding algorithms that assume a uniform distribution over all positions in a sequence, meaning that the likelihood of a fragment being a valid motif is independent of where it occurs in the sequence. While this assumption is valid in the general case (i.e. when the context is unknown), the fact that RBS's tend to be at a reasonably conserved range of positions from gene-starts allows us to impose a stricter requirement. To do that, we assume that there is a distribution of probabilities, M_{pos}, for a motif to start in a given position defined over $L - W + 1$ possible starting positions. Collectively, the models are designated by $\lambda = (M_{motif}, M_{bgd}, M_{pos})$.

Now, the probability of the alignment of all motifs (putative functional sites) along with their flanking background sequences can be expressed as follows:

$$\begin{aligned} P(\boldsymbol{a}|\boldsymbol{S}, \lambda) &= \prod_{n=1}^{N} P(a_n|S^{(n)}, \lambda) \\ &= \prod_{n=1}^{N} P(S^{(n)}_{1...a_{n-1}}|M_{bgd}) P(S^{(n)}_{a_n...a_{n+W-1}}|M_{motif}) P(S^{(n)}_{a_n+W...L}|M_{bgd}) P(a_n|M_{pos}) \end{aligned}$$

At each iteration, the distribution from which a new motif position l in sequence n is sampled is defined by the normalization with respect to the sequence having no motif at all (i.e. generated entirely by the background model). This is defined as

$$A^{(n)}_l = \frac{P(l|S^{(n)}, \lambda)}{P(l|S^{(n}, M_{bgd}, M_{unipos})} = \frac{P(S^{(n)}_{l...l+W-1}|M_{motif})}{P(S^{(n)}_{l...l+W-1}|M_{bgd})} \frac{P(l|M_{pos})}{P(l|M_{unipos})} \tag{A.9}$$

where M_{unipos} is a uniform distribution over all positions. The overall model λ is updated each time a new motif position is sampled.

This process is repeated in iterations, and favors the assignments of motif positions that maximize the alignment score F: the log of the probability of aligned sequences with given motif positions, computed using the M_{motif}, M_{bgd} and M_{pos} models divided by the

probability of the same sequences computed solely by the M_{bgd} and M_{unipos} models. This is defined as

$$F = \log \frac{P(\boldsymbol{a}|S, \lambda)}{\prod_{n=1}^{N} P(a_n|S^{(n)}, M_{bgd}, M_{unipos})} = \sum_{i=1}^{W} \sum_{j=1}^{J} c(i,j) \log \frac{M_{motif}(i,j)}{M_{bgd}(j)} + \sum_{l=1}^{L} c(l) \log \frac{M_{pos}(l)}{M_{unipos}(l)} \tag{A.10}$$

This is equivalent to the KL divergence between the motif and background models (including the position models), i.e.

$$F = \mathrm{KL}(M_{motif}||M_{bgd}) + \mathrm{KL}(M_{pos}||M_{unipos}) \tag{A.11}$$

Here J is the size of the alphabet (e.g. 4 in the case of nucleotides). The function $c(i,j)$ is the number of times element j appears in position i of the predicted motifs, and $c(l)$ is the number of times motifs are located at position l. Similarly, $M_{motif}(i,j)$, $M_{bgd}(j)$, and $M_{pos}(l)$ are the probabilities of symbol j in the motif at position i, symbol j in the background, and the motif start location l, respectively. After each K iterations, it is checked if a shifted form of the motif model results in a larger score F. This involves shifting all motifs by a small set of integer values $z \in [-2, 2]$, and comparing the F_z score of the new alignment for each of these values. This allows the algorithm to escape local optimums and, thus, construct alignments with higher scores.

A.5 Results

A.5.1 Gene finding accuracy evaluation

We demonstrated in several tests that, on average, GeneMarkS-2 is a more accurate tool than the current frequently used gene finders. Particularly, GeneMarkS-2 made fewer false

Table A.1: Statistics of false negative (panel A) and false positive gene predictions (panel B) observed in tests on 54 genomes containing proteomic validated genes and on 145 genomes with genes validated by orthologues in COGs.

Algorithm	Missed MS confirmed genes (from 89,466)	Missed COG genes (from 287,237) (not MS)
GeneMarkS	376	1,467
Glimmer3	496	1,990
Prodigal	217	1,389
GeneMarkS-2	**181**	**1,147**

Algorithm	False predictions overlapping MS confirmed genes	False predictions overlapping COG genes (not MS)
GeneMarkS	352	2,046
Glimmer3	921	6,435
Prodigal	211	1,339
GeneMarkS-2	**114**	**932**

Table A.2: Panel A: Counts of genes missed by a particular tool (false negatives) among 341,486 COG genes annotated in 145 genomes. The counts are given in five length bins. Panel B: Counts of false positive predictions made in 144 simulated genomic sequences made from 144 original genomes where annotated intergenic regions were replaced by artificial non-coding sequence (see text). The numbers of false predictions were sorted by length in the same way as in Panel A.

A	**Bins (nt):**	**<150**	**150-300**	**300-600**	**600-900**	**>900**	**Total**
Algorithm	COG genes	362	13,985	65,948	83,745	177,446	341,486
		Missed annotated genes (FN)					
GeneMarkS		136	**494**	434	192	296	1,552
Glimmer3		**66**	678	1,170	341	323	2,578
Prodigal		161	639	417	92	78	1,387
GeneMarkS-2		132	596	**370**	**76**	**69**	**1,243**

B	**Bins (nt):**	**<150**	**150-300**	**300-600**	**600-900**	**>900**	**Total**
Algorithm		*False positives (FP) in simulated sequence*					
GeneMarkS		3,366	5,113	1,230	177	94	9,980
Glimmer3		17,446	5,044	1,299	228	136	24,153
Prodigal		4,525	5,321	1,453	419	135	11,853
GeneMarkS-2		**792**	**1,541**	**601**	**137**	**77**	**3,148**

negative and false positive errors in predicting genes validated by mass-spectrometry and COG annotation (Table A.1). Also, the numbers of false positive predictions made by GeneMarkS-2 in simulated non-coding sequences were significantly smaller than the numbers observed for other tools (Table A.2).

The array of atypical models employed in GeneMarkS-2 improved the prediction of horizontally transferred (atypical) genes. In our observations, the deviation of GC composition of atypical genes from the genome average could be as large as 16% (e.g. the 798 nt long E. coli gene *b0546* characterized as *DLP12* prophage, with GC content 36%

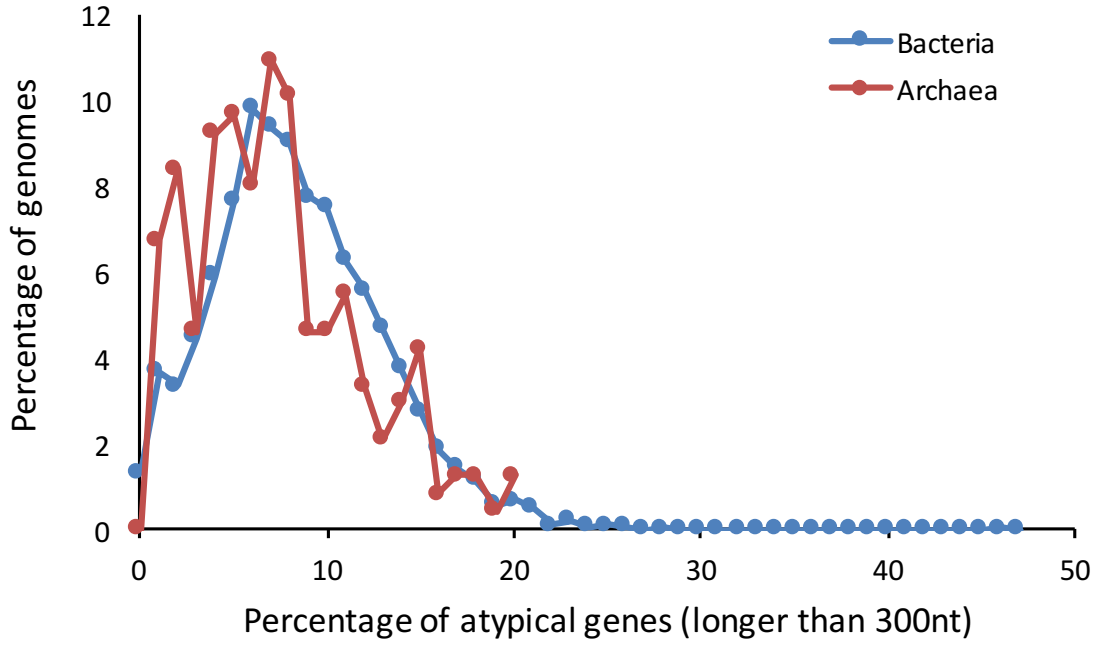


Figure A.2: Distributions of the percentage of predicted 'atypical' genes in archaeal and bacterial genomes.

compared to the 52% GC content of the bulk of E. coli genes). The GC content of atypical genes is frequently lower than the GC content of 'typical' ones (Fig. S13). Also, the 'atypical' genes with large GC content deviations are expected to appear more frequently in high GC genomes given the larger space for downward variation. All in all, atypical genes may constitute a significant fraction of the whole gene complement (e.g. about 15% of genes in the E. coli genome [37]). In our analysis of the ~5,000 genomes, we found that the distribution of the fraction of predicted atypical genes in prokaryotic genomes are rather similar between archaea and bacteria, with an average of about 8-9% (Figure A.2).

A comparison of the sets of COG annotated genes missed by the three gene finders (Fig. S14) shows that atypical genes, genes predicted by atypical models of GeneMarkS-2, constituted 30% of 780 (534+246) genes missed by Prodigal and 42% from 1605 (1359+246) genes missed by Glimmer3. Both Prodigal and Glimmer3 employ a single model of protein-coding regions. We argue that the more accurate prediction of atypical genes by

GeneMarkS-2 makes a compelling argument in favor of the use of multiple models of protein-coding regions.

One of the features of GeneMarkS-2 is the ability to characterize atypical genes as bacterial or archaeal due to the division of the atypical models into distinct bacterial and archaeal types [38]. The insights into the possible origin of atypical genes (likely horizontally transferred) could be particularly useful for genomes of thermophilic bacteria and mesophilic archaea.

A.5.2 Error rates in prediction of protein-coding genes (all but the 5' end)

Gene predictions made by GeneMarkS, Glimmer3, Prodigal, and GeneMarkS 2, run with default settings, were compared with (i) annotation of the MS or "proteomics" validated genes and (ii) the COGs validated genes. In the 54 genomes, there were ~89,500 proteomics supported genes (psORFs); in the 145 genomes, there were ~341,486 genes in total, 287,237 of which did not overlap with the proteomics validated genes (Table A.2).

In the set of 54 genomes, we observed that GeneMarkS 2 missed 181 psORFs out of 89,466, the least number of false negative errors made by the tested tools (Table A.1).F At the same time, GeneMarkS-2 made the least number of false positive predictions, 114. A predicted gene was judged as false if more than 30% of its length overlapped with a psORF located in one of five other frames.

The comparison of the predictions with the COG validated genes demonstrated higher accuracy of GeneMarkS-2 as well. The new tool missed the lowest number of COG genes, 1,147, followed by Prodigal with 1,389. Notably, the rate of missed COG genes by any single gene finder was less than 1% (Table A.1). Counting false positives (identified also as ones with prohibitively long overlaps with verified genes) has shown that GeneMarkS-2 made 932 false predictions, a significantly smaller number than the ones made by the other gene finders (Table 10). Note that the COG validated genes identical to the "proteomics" supported genes were excluded from the second test.

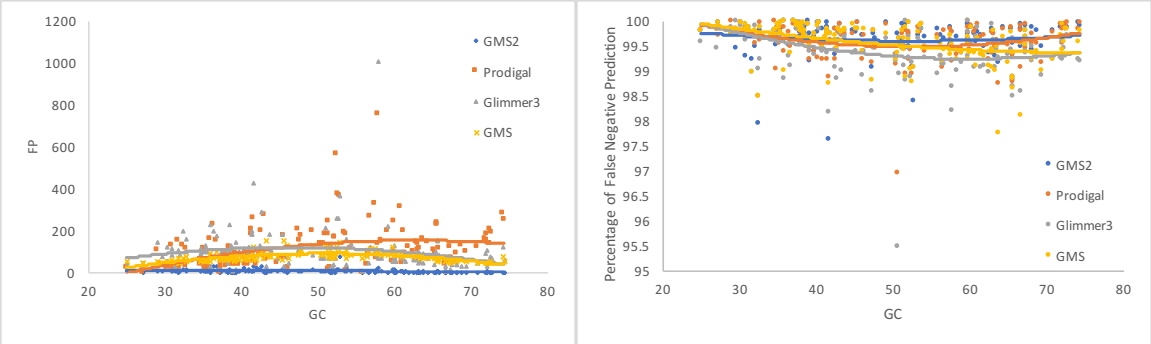


Figure A.3: The dependence of false positive and false negative rates on the genome's GC.

We specifically looked into the distribution of the two types of errors with respect to the gene length (Table A.2). Among all COG annotated genes Glimmer3 missed the least number of short genes (in 90-150 *nt* range) in comparison with the other tools. In the next bin, 150-300 *nt*, GeneMarkS did show the best result. We observed that GeneMarkS-2 missed the least numbers of COG genes with length >300 *nt* and made the least total count. Its performance was the least dependent on genome GC content (Figure A.3).

Since it turned out that the false positives identified by their too long overlaps with validated genes (confirmed by proteomics or by COG annotation) occurred in rather small numbers (Table A.1), we attempted to offer more substantial statistics by adding tests on sets of synthetic sequences.

A.5.3 False positive predictions in synthetic sequences

We estimated the false positive rates on the three sets described above. First, we ran the four gene finders on the 144 constructs where the intergenic sequences were replaced by the same length synthetic non-coding sequences while the annotated genes remained in place. The predicted genes with 3' end not matching annotation were considered as false positives (Table A.2). Notably, the number of false negatives in these experiments was observed to be of the same order as in the runs of the gene finders on the original genomes, i.e. ~1% of the number of annotated genes. Second, we ran the four gene finding tools trained on 145 complete genomes on Set 2, the 145 sequences of length 100,000 *nt*, the experiment

repeated 10 times (data not shown). The first and second types of experiments had the advantage of keeping the trained parameters consistent with the non-perturbed features of the genome. However, the gene finders were not adapted to the simulated non-coding sequences. Therefore, we ran the gene finders in full cycle, training and prediction, on Set 3 where predictions made in the 100,000 *nt* extended artificial portion of each genome were counted as false positives (data not shown).

The results of all the three types of tests described above were favorable for GeneMarkS-2. Further analysis demonstrated that reduction of false positives GeneMarkS-2 in comparison with GeneMarkS was mainly due to the improvement of the parameterization of the atypical models. Glimmer3 has made frequent false positives predictions of the short length ($<$150 *nt*). This outcome is, arguably, the cost of the higher than other gene finders sensitivity in this length range (Table A.2).

Prodigal generated false positive predictions of rather long length as the algorithm settings give high weights to longer ORFs. Subsequently it leads to increase of false positives in genomic sequences with high GC content (Figure A.3) where longer ORFs appear more frequently than in low GC genomes.

All over, in all the five approaches of the assessment of false positive rates (two in natural sequences and three in artificial ones) we saw that GeneMarkS-2 demonstrated the best performance. Notably, it yielded the lowest numbers of false positives in all the length intervals.

A.5.4 Variations in width of RBS model

To support the default motif width (6*nt*) used for RBS motifs, we tested out varying widths (from 5 to 10) and showed the overall gene-start accuracy on the genomes with experimentally verified starts. As is shown in Table A.3, the motif width generally has little effect on the start accuracy, peaking at widths 6 and 9. Furthermore, we showed that in the case of *E. coli*, the RBS motifs wider than 6*nt* did not seem to capture additional information missed

Table A.3: The dependence of the gene start accuracy on the RBS motif width; computed over the set of seven genomes with experimentally verified starts.

Species	Gene-start model type	# of verified gene starts	Width 5	Width 6	Width 7	Width 8	Width 9	Width 10
*A. pernix**	A	130	124	126	125	**127**	126	**127**
D. deserti	C	384	369	369	370	369	**371**	369
E. coli	A	769	742	740	741	743	742	**745**
*H. salinarum**	D	530	**524**	523	522	521	522	523
M. tuberculosis	C	701	632	**635**	631	**635**	634	632
*N. pharaonis**	D	315	**312**	**312**	311	311	**312**	310
Synechocystis	X	96	90	**92**	91	89	90	88
(*archaea)	**Total**	2,925	2,793	**2,797**	2,791	2,795	**2,797**	2,794

by the 6*nt* motif model, which led us to settle on 6*nt* as the default value.

A.5.5 Effects of the Addition of the Length Distribution Terms in GibbsL

A canonical form of the Shine-Dalgarno (SD) ribosomal binding site (RBS) motif is represented as AGGAGG. The abundance of G's in that sequence indicates that random sequences with high GC are more likely to exhibit similar hexamers than low GC sequences.

Consider, then, the task of searching for an RBS motif in upstream intergenic regions of length 40 nt in high GC genomes. Suppose that one of these upstream sequences has an RBS motif at a distance 6 nt from the gene start. Also assume, given it's high GC nature, that that sequence has a rich-in-G hexamer located further down at a distance of 32 (way beyond the expected location of an RBS).

If the motif search algorithm does not take distance into account, then it will equally likely choose between these two candidates, and may choose the (incorrect) farther hexamer over the real RBS motif. In the case of GibbsL, however, the distribution over RBS positions (derived from the remaining upstream sequences) is used to handle these quasi tie-breakers by preferring motifs localized in the same region. Figures S9 an S10 clearly shows that this effect is growing as the GC of the sequence increases (in the case of RBS search).

A.6 Supplementary Figures

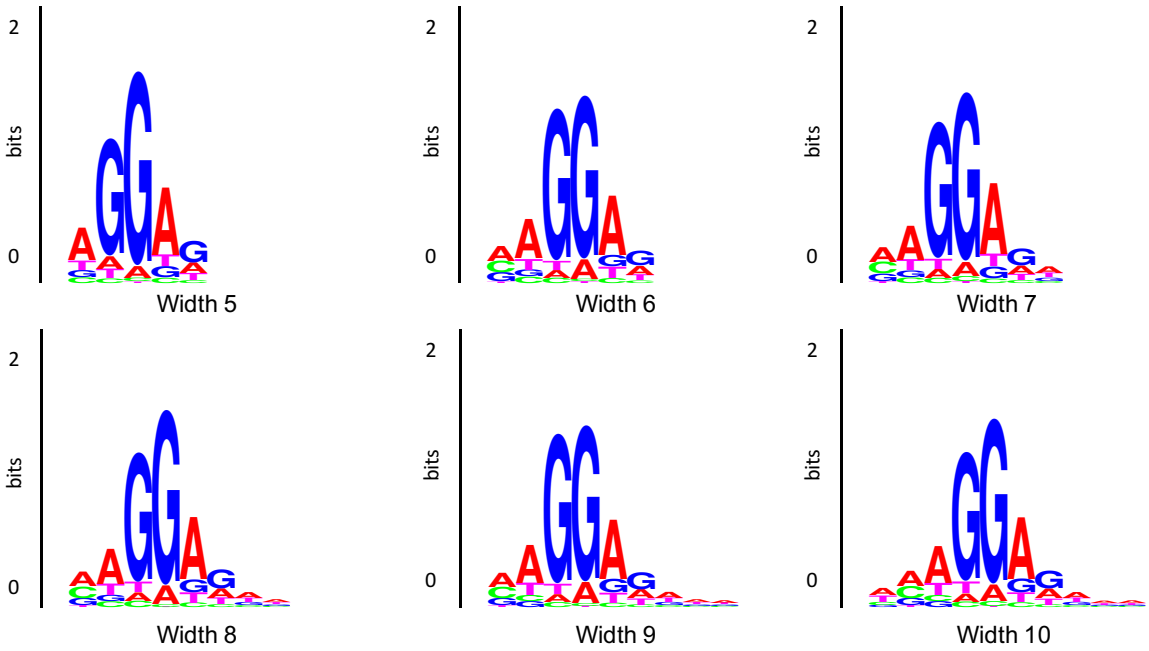


Figure A.4: The motif logos of different widths derived for *E. coli* by GeneMarkS-2.

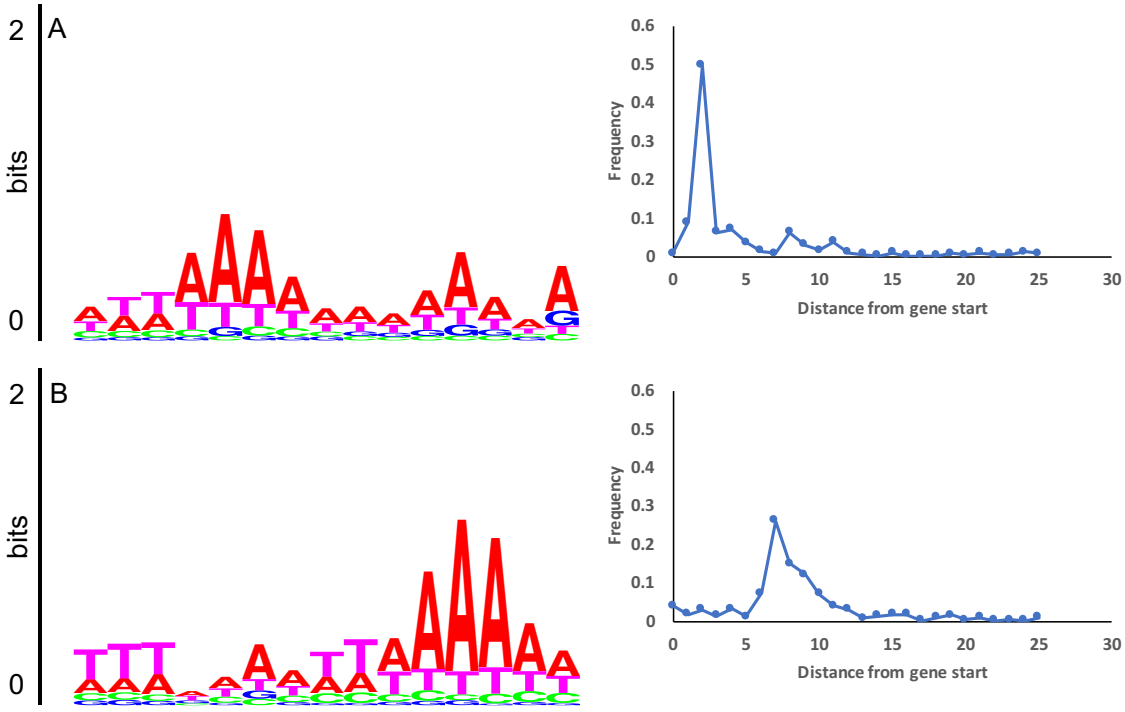


Figure A.5: The motif logo and the spacer length distribution for a 15*nt* motif signal, for two group B genomes: (A) *Bacteroides vulgatus* and (B) *Flavobacterium johnsoniae*. Note that while the logo is shown in the 5' to 3' direction of the nucleotide sequence with the gene start assumed to be on the right, the spacer length distribution is shown in positive "distance scale" instead of the negative scale of "biological" coordinates. Distances from the start were computed to the 5' end of the motifs shown in Panels A and B.

APPENDIX B

STARTLINK

B.1 Accuracy of Combined Predictors

The derivations of (Equation 4.1) under the three conditions of randomness, independence, and full dependence, is as follows. The first expansion (irrespective of condition) is

$$
\begin{aligned}
& p(y = s | x_1 = y, x_2 = y) \\
&= \frac{p(x_1 = y, x_2 = y, y = s)}{p(x_1 = y, x_2 = y)} \\
&= \frac{p(x_1 = y, x_2 = y, y = s)}{p(x_1 = y, x_2 = y, y = s) + p(x_1 = y, x_2 = y, y \neq s)} \\
&= \frac{1}{1 + \frac{p(x_1=y,x_2=y,y\neq s)}{p(x_1=y,x_2=y,y=s)}} \\
&= \frac{1}{1 + \frac{p(y\neq s)\ p(x_2=y|y\neq s)\ p(x_1=y|x_2=y,y\neq s)}{p(y=s)\ p(x_2=y|y=s)\ p(x_1=y|x_2=y,y=s)}} \\
&= \frac{1}{1 + \frac{p(y\neq s)\ \mathrm{Err}(A_2)\ p(x_1=y|x_2=y,y\neq s)}{p(y=s)\ \mathrm{Acc}(A_1)\ p(x_1=y|x_2=y,y=s)}}
\end{aligned}
\tag{B.1}
$$

Note that assuming uniform priors, we have

$$
\begin{aligned}
P(y = s) &= \frac{1}{|C|} \\
P(y \neq s) &= \frac{|C| - 1}{|C|}
\end{aligned}
$$

Specifically, the prior over $p(y \neq s)$ assumes that when an algorithm makes a wrong prediction, it is equally likely to choose any of the false candidates. In reality, this may or may not be true, especially if algorithms tend to choose false starts closer to the true start. This

is important especially under conditions where algorithms are somewhat dependent. However, given that we have no knowledge of A_1 and A_2, the most straightforward approach is to assume an uninformed prior.

Therefore, we get

$$\begin{aligned} & p(y = s|x_1 = y, x_2 = y) \\ & = \frac{1}{1 + (|C| - 1)\frac{\mathrm{Err}(A_2)\ p(x_1=y|x_2=y,y\neq s)}{\mathrm{Acc}(A_1)\ p(x_1=y|x_2=y,y=s)}} \end{aligned} \tag{B.2}$$

Case 1: Algorithms A_1 and A_2 are uniform, random selectors.

Since A_1 and A_2 are completely random, then they are (by definition) independent. This implies that

$$\begin{aligned} p(x_1 = y|x_2 = y, y = s) = p(x_1 = y|y = s) = \frac{1}{|C|} \\ p(x_1 = y|x_2 = y, y \neq s) = p(x_1 = y|y \neq s) = \frac{|C| - 1}{|C|} \end{aligned}$$

Therefore, we have

$$\boxed{p(y = s|x_1 = y, x_2 = y) = \frac{1}{|C|}}$$

Case 2: Algorithms A_1 and A_2 are completely **independent**

Again, since A_1 and A_2 are independent, we have

$$\begin{aligned} p(x_1 = y|x_2 = y, y = s) = p(x_1 = y|y = s) = \mathrm{Acc}(A_1) \\ p(x_1 = y|x_2 = y, y \neq s) = p(x_1 = y|y \neq s) = \mathrm{Err}(A_1) \end{aligned}$$

Therefore, we have

$$\boxed{p(y = s|x_1 = y, x_2 = y) = \frac{1}{1 + (|C| - 1)\frac{\mathrm{Err}(A_2)\,\mathrm{Err}(A_1)}{\mathrm{Acc}(A_2)\,\mathrm{Acc}(A_1)}}}$$

Case 3: Algorithms A_1 and A_2 are completely **dependent**

Again, since A_1 and A_2 are independent, we have

$$p(x_1 = y | x_2 = y, y = s) = 1$$
$$p(x_1 = y | x_2 = y, y \neq s) = 1$$

Therefore, we have

$$\boxed{p(y = s | x_1 = y, x_2 = y) = \frac{1}{1 + (|C| - 1)\frac{\mathrm{Err}(A_2)}{\mathrm{Acc}(A_2)}}}$$

B.2 Effect of Kimura distance on StartLink

StartLink uses Kimura distances as a metric to filter out very close or distant relatives. We tested several alternative evolutionary distance metrics, including (non-)synonymous substitution rates, amino acid identity, etc. but they either performed equally well or slightly worse at providing StartLink with a good set of orthologous sequences (data not shown). Overall, we found that these metrics provide a way to remove very close/distant relatives, but that additional filtering (such as analyzing the gaps in the MSA incurred by each sequence (see main algorithm)) might still be needed to clean up some instances.

Figure B.1 shows the effect of varying the maximum Kimura threshold from 0.2 to 0.8, while fixing the minimum threshold to 0.1. StartLink's sensitivity is rather stable as the maximum Kimura value increases. On the other hand, the coverage rate fluctuates more in some genomes. This is because as more sequences with larger Kimura (and thus, more mutations) are potentially included in the MSA, StartLink will have a difficulty making a prediction on the MSA.

Specifically, *N. pharaonis* and *H. salinarum*'s coverage rates suffer when the maximum Kimura threshold below is 0.4. Note, however, that the available genomes under the entirety of Archaea (used as these genomes' ancestral clade) is quite small, especially when

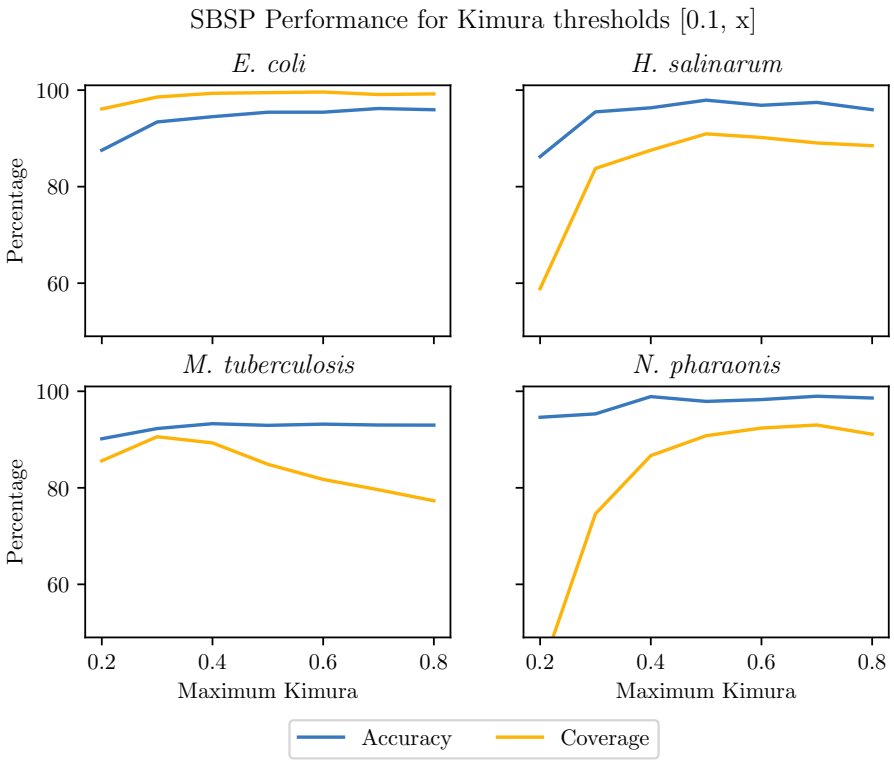


Figure B.1: The effect of changing the maximum Kimura threshold on StartLink's sensitivity and coverage rates. The minimum Kimura threshold is fixed to 0.1, and $x \in \{0.2, 0.3, ..., 0.8\}$.

compared to *Enterobacterales* and *Actinobacteria*. This means that, in general, it is less likely to find close sequences, which directly impacts the coverage rate.

On the other hand, *M. tuberculosis*'s coverage rate decreases as the maximum threshold increases past 0.4. A direct inspection of the MSAs shows that they appear to have more mutations around the gene-start region than, for example, in *E. coli* for the same range of Kimura distances. It is unclear whether this is due to a bias in the database or a biological feature. That said, StartLink's sensitivity rate remains constant, meaning that for such alignments, it does not make a prediction.

Similarly, Figure B.2 fixes the maximum Kimura threshold to 0.5 and varies the minimum threshold from 0.001 to 0.4. Overall, this has a lower effect on the coverage and sensitivity rates than in the previous case. We see an initial drop in the case of *E. coli* when the minimum threshold is close to 0.001. In this range, we observed more orthologs with

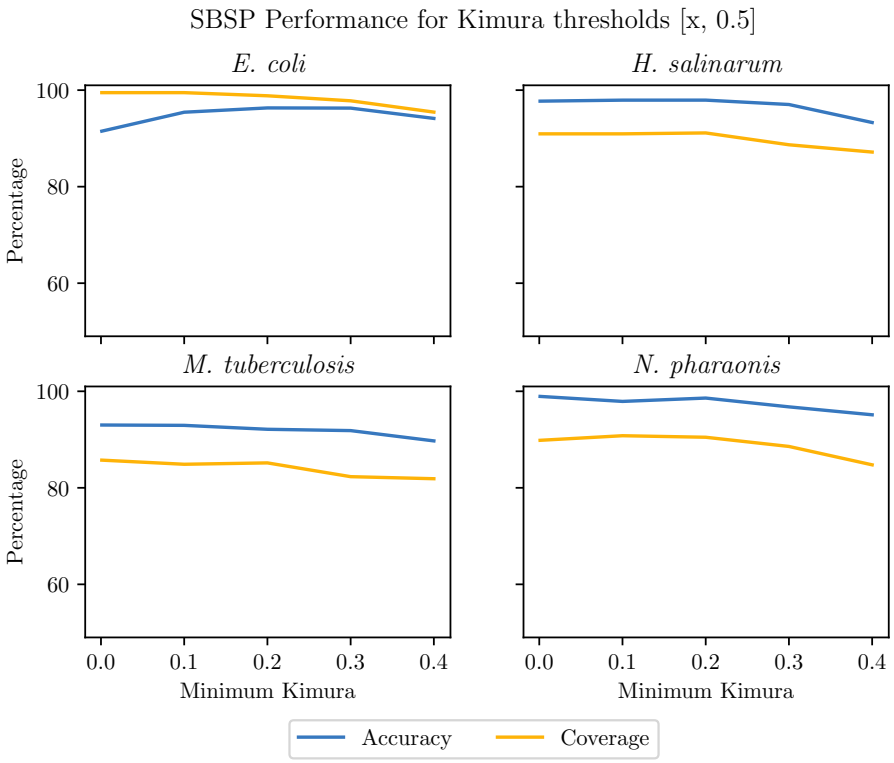


Figure B.2: The effect of changing the minimum Kimura threshold on StartLink's sensitivity and coverage rates. The maximum Kimura threshold is fixed to 0.5, and $x \in \{0.001, 0.1, 0.2, 0.3, 0.4\}$.

high nucleotide-level identity rate (to the query gene) than in the other three genomes. It is again unclear whether this stems from a bias in the database or from some biological properties, but the overall reduction in sensitivity is still small.

It is important to note that for all the above experiments, the sensitivity of StartLink+ remains consistently high and largely unaffected, while the coverage rate mirrors that of StartLink (since it is directly dependent on it) (data not shown). In other words, our selection of Kimura distances does not need to be fine tuned beyond what has been done. This is fortunate, given that we do not want to overfit the little ground-truth data available to us.

Finally, we test StartLink on small ranges of Kimura, specifically $[0.001, 0.1], [0.1, 0.2], \ldots [0.7, 0.8]$. This forces StartLink to operate exclusively in regions of very close and very distant orthologs, and allows us to see a more detailed view of the effect of Kimura on the algorithm. Figure B.3 shows the effect these ranges on the sensitivity and specificity.

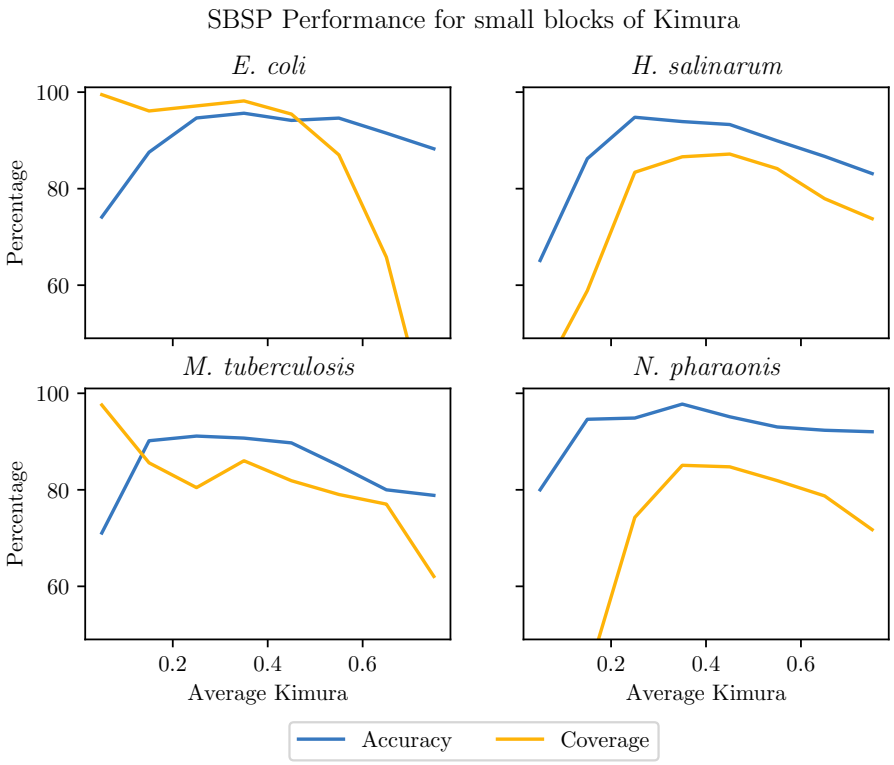


Figure B.3: The performance of StartLink on small intervals of Kimura ranges: $[0.001, 0.1]$, $[0.1, 0.2]$, $[0.2, 0.3]$ $\ldots$ $[0.7, 0.8]$. The x-axis shows the mean Kimura of a block; e.g., for range $[a, b]$, the average is $(b + a)/2$.

The effect here is more significant, especially at the extreme blocks ($[0.001, 0.1]$, $[0.7, 0.8]$). The sensitivity rate is still rather stable, with the biggest impact being in the range $[0.001, 0.1]$. The coverage rate, however, is more striking. First, note that the coverage rate behaves differently in *E. coli* and *M. tuberculosis* compared to the two archaea. Specifically, it starts out high in both *E. coli* and *M. tuberculosis*, and then decreases as we move beyond the $[0.4, 0.5]$ range. The drop is especially strong in *E. coli*, where StartLink struggles to make predictions for ranges [0.6, 0.7] and above (although still maintaining a high and stable sensitivity rate).

On the contrary, the coverage rate is low for *H. salinarum* and (especially) *N. pharaonis* for the [0.001, 0.1] and [0.1, 0.2] ranges. It then rises quickly and stabilizes somewhat, before dropping a little in the higher ranges. The reasons for this are, in part, due to how sparsely populated the database of Archaea is, with just over 1,000 genomes for the entire

clade, compared to over 6,311 and 8,097 genomes for *Enterobacterales* and *Actinobacteria*, respectively, which are much lower nodes in the taxonomy tree. In fact, *E. coli* and *M. tuberculosis* each has $1,000+$ sequenced genomes just from their own species.

Therefore, this difference in effect is likely to only be a result of the underlying database population, and may disappear once the database is populated more uniformly.

B.3 Features around gene-starts

The StartLink algorithm relies on simple metrics (such as an identity measure of coding region conservation) computed in non-coding and coding regions and around true and false candidate starts. These simple metrics work given the prior Kimura filtering of removing close and distant sequences from the multiple sequence alignment (MSA).

In this section, we show what the scores used by StartLink look like in different parts of the multiple sequence alignment. We take genes whose starts are experimentally verified and run them as queries through the standard StartLink algorithm. This gives us a multiple sequence alignment for each query gene. The goal is to show how the two scores defined in (Equation 4.10) and (Equation 4.9) behave in different regions of the MSA (specifically, upstream of, at, and downstream of the verified start).

First, we emphasize the following: this analysis is done from the perspective of StartLink to show how the algorithm behaves. This means that conditions that StartLink uses, such as only extracting sequences up to their LORF and filtering by Kimura, are inherently part of the analysis. For example, if a query's verified start is at the LORF, then the scores are not computed for the non-coding region, since StartLink will not search for a gene-start there.

B.3.1 Conservation block

We start by analyzing the conservation block score S_{blk} defined in (Equation 4.9). For each MSA, we take blocks (of width 10 *aa*) in four different regions: 20 *aa* upstream of the verified start (Up-far), just upstream of the verified start (Up-close), just downstream of

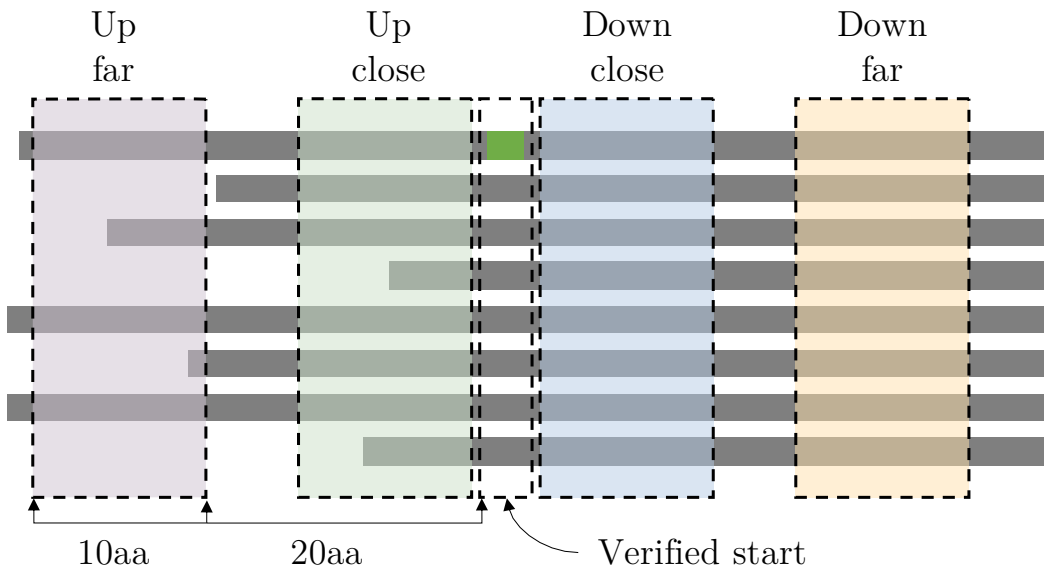


Figure B.4: An illustration of the four selected regions of the MSA: close/far upstream and downstream regions. The distant regions are 20 *aa* from the verified start, and all regions have a width of 10 *aa*.

it (Down-close), and 20 *aa* downstream of the verified start (Down-far). The regions are shown in Figure B.4.

Figure B.5 shows the distribution of scores for each block region. As expected, the upstream (non-coding) regions have a much lower conservation score than the downstream (coding) regions. Mutations are more likely to happen in non-coding regions than in coding regions, and given that these sequences are already at a significant evolutionary distance from each other, the number of mutations in the non-coding region is expected to not be negligible. Second, given that sequences are extracted up to their LORF, this region of the MSA can have a large number of gaps which decreases overall score. This works to our advantage, since we want StartLink to skip over such regions. Notice that the specifics of the StartLink setup (Kimura filtering, LORF extraction, etc...) is what allows us to use a simple majority-vote metric based on identity conservation. This lets StartLink easily skip over non-coding regions, making this a reliable metric for ensuring that StartLink does not detect a conserved block in the non-coding region.

On the opposite end, the two regions downstream of the verified start both show rela-

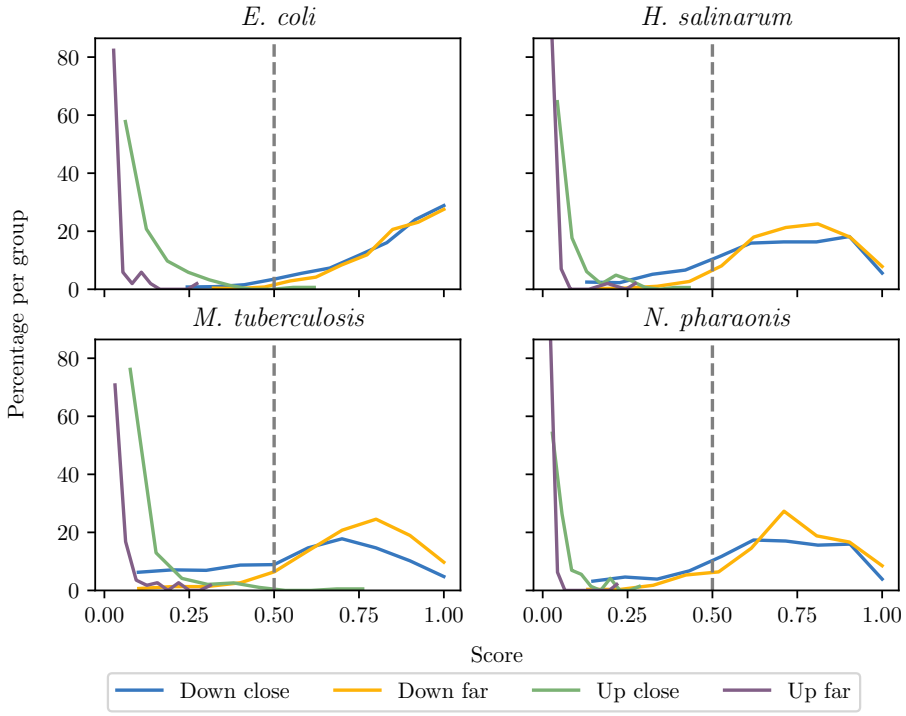


Figure B.5: Distribution of block conservation scores in regions around verified starts.

tively high conservation scores. Interestingly, Down-far has a higher conservation scores (on average) than Down-close. This is especially apparent in *M. tuberculosis*. This indicates that non-synonymous mutations just downstream of the gene-start may be more likely to occur than in a random section deeper within a gene. This may be a reason why evolutionary distance metrics (which are typically computed on the entire gene) may not be capable (on their own) of selecting an optimal set of sequences for StartLink's gene-start search algorithm.

B.3.2 Gene-Start Identity Score

Similar to the above analysis, we analyze the effectiveness of the 5' score ((Equation 4.10)) at separating the true from the false starts. One challenge here is that there are two types of false starts that exhibit different properties.

In an ideal case, the true start will experience a very high 5' identity score (as defined

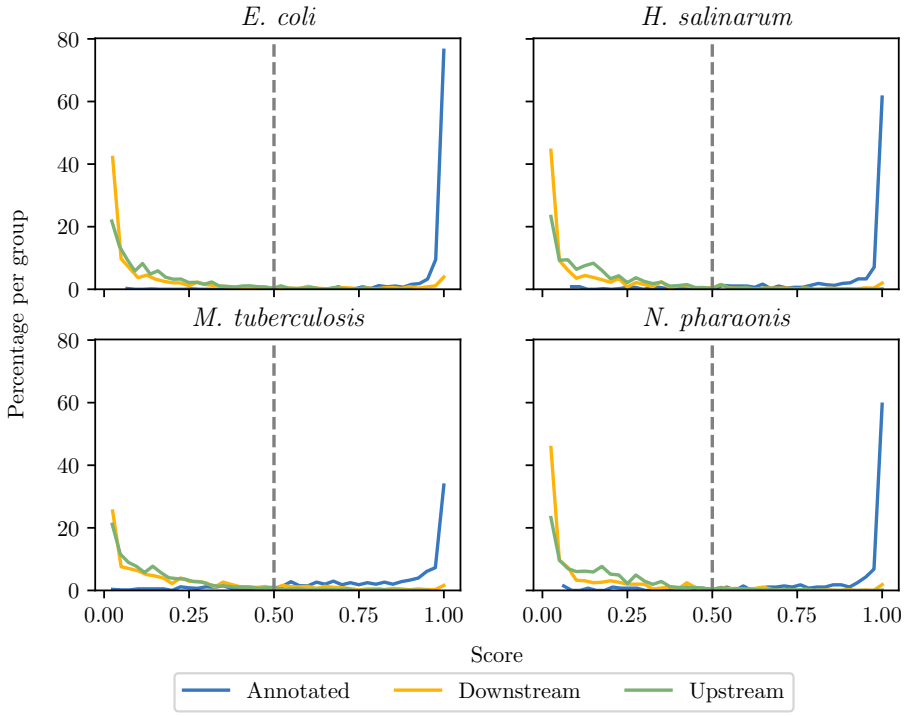


Figure B.6: Distribution of 5' identity for verified starts, and upstream and downstream false 5' candidates.

by (Equation 4.10)), due to orthologs preserving the location of the 5' end. The upstream (false) candidate tends to have a low conservation score, due to it being in the non-coding region (with effects similar to what was discussed in the previous section). On the other hand, the false candidates downstream of the verified start can also show high identity rates, since this region is typically conserved across orthologs. This was the reason why the penalty for synonymous codons was added in (Equation 4.10).

Figure B.6 shows how the 5' score separates the annotated (in this case, verified) gene-starts from the false candidates. The gap between the groups is strikingly wide, meaning that the 0.5 threshold is a pretty reasonable, non-overfitting value.

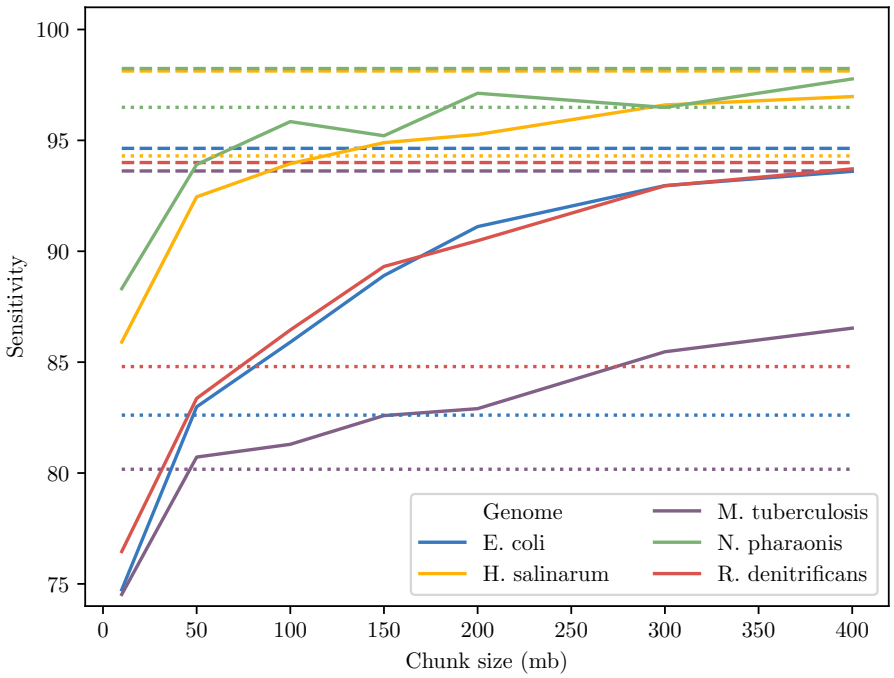


Figure B.7: The sensitivity rate of GeneMarkS-2 when genomes are broken into smaller fragments. The dashed and dotted lines show the corresponding sensitivity of StartLink and MetaGeneMark, respectively, for each genome.

B.4 Metagenomic gene-start prediction

Standard self-training *ab initio* predictors usually require a sequenced genome for training their parameters. In many cases, however, such as metagenomic assembly, only part of the genome can be retrieved. In this case, tools such as GeneMarkS-2 can suffer due to insufficient training data. In fact, on small scales such as these, GeneMarkS-2 can reduce to a run of MetaGeneMark [38], which relies on previously built, non-native gene models that exclude motif information.

Figure B.7 shows a simulated experiment of what GeneMarkS-2's 5' sensitivity would be if only part of the genome could be sequenced. Since StartLink only requires the gene itself, it's accuracy is stable irrespective of the length of the genomic fragment (as long as the gene itself is within the sequenced fragment).

We tested GeneMarkS-2's performance as a function of the genome length. We did this by cutting the genomes into smaller chunks, running GeneMarkS-2 separately (and forcing it to train a native model) on each, and then recombining the results to compute overall sensitivities. As shown in Figure B.7, the performance of GeneMarkS-2 starts to degrade rapidly when the input fragment size goes below the 200K range. And at 10K, we see a drop all the way down to 75% on *M. tuberculosis*, *R. denitrificans*, and *E. coli*, 96% on *H. salinarum* and 88% on *N. pharaonis*. This is in contract to StartLink, which doesn't depend on the genome size at all, and maintains an average accuracy of 95% on all genomes, with the lowest being *M. tuberculosis* at 93.62%.

This is not meant to be a comprehensive analysis of metagenomic gene-start prediction. However, the advantage of similarity-based approaches is their ability to make predictions without relying on full-genome assembly. That said, these approaches fail for genes whose orthologs have not yet been annotated in the database. As such, non-similarity-based approaches will be preferred in such cases.

B.5 StartLink: Technical Details and Optimizations

In this section, we discuss some under-the-hood details of the selection process of target sequences in StartLink, focusing on items that can affect the overall performance and runtime. Before we begin, we state the ultimate goal of this selection process by describing the desired properties of the final selected set of sequences. We then describe the selection process and discuss optimizations and shortcuts made to reduce the overall runtime.

In StartLink, target sequence selection is a multi-step process. We start with a search of protein sequence queries on a database and then filter out unwanted hits. Given that our initial databases are large, we use Diamond BLASTp, which provides a faster version of the BLASTp search algorithm. This search typically returns a thousands of hits per query, and the goal is to select N (in our case, $N = 50$) target sequences (per query) that will be used by StartLink.

Ultimately, we want to choose a set of sequences that, when aligned, would allow us to find the start of the query gene easily. For our purposes, we want this set S (a query and (maximum) N targets) to satisfy the following pairwise constraints:

$$\mathrm{d_k}(s_1, s_2) \in [0.1, 0.5] \qquad \forall s_1, s_2 \in S, s_1 \neq s_2 \tag{B.3}$$

This means that no pair of sequences in this set should have a distance smaller than 0.1 or larger than 0.5.

The most straightforward approach to accomplishing this is to compute all pairwise Kimura distances between query and target sequences, and then randomly select N that exist in the desired range. However, Kimura distance calculation requires a global alignment of two sequences, which in this case would be very costly (given the large initial number of sequences).

B.5.0.1 Avoiding Global Alignment

Therefore, the first order of approximation is to use the already-existing local alignments between a query and its targets (generated by the BLASTp search), and immediately compute approximate Kimura values. In our experiments, we found no tangible difference in StartLink's performance on the verified set based on whether global or local alignment was used in the Kimura computation.

Note, however, that these local alignments are only provided between a query and a target, and not between pairs of target sequences. Furthermore, there can still be tens of thousands of targets per query, and going through them all would result in many unnecessary computations.

B.5.0.2 Avoiding all-pairs target alignments

We can avoid (at this stage) from directly comparing target sequences to each other by observing how they compare to the query. This step exploits the fact that two very similar target sequences should have similar distances to the query. As such, if the distance between two targets t_1 and t_2 and the query is roughly the same (i.e. $d_k(q, t_1) \approx d_k(q, t_2)$)), then we remove one of them from the list of candidates under the assumption that they are more likely to be as similar to each other. We deem two distances as approximately the same if they are equal up to the fourth decimal place.

It's important to note that this heuristic is not optimal, however. In particular, it doesn't hold in the opposite direction; i.e., two target sequences with similar Kimura distances to the query does not imply that the targets are similar to each other. That said, we've found that this step slightly improves performance because we are less likely at this stage to import multiple target sequences that are very similar to each other, which end up biasing the alignment.

B.5.0.3 Avoiding analyzing the large number of target sequences

We first note that hits retrieved from BLASTp are ordered by e-value. In practical terms, this means that they are ordered in terms of some similarity measure to the query sequence. It also means that they are roughly ordered by Kimura distance; we say "roughly" because the ordering is not exact, but it does not affect the final performance.

Given this list, we want to find the position of the sequence after which no sequence exists with $d_k \in [a, b]$. If the list was perfectly sorted, this task would boil down to finding the first sequence with $d_k > b$. Given that the ordering is not exact, however, we take some measures to decrease the likelihood that our heuristic removes sequences within the range $[a, b]$.

First, if the total number of target sequences for a query is less than 2,000, we skip this filtering step since the number of computations needed is not large. Second, instead of

looking for the first sequence with $d_k > b$, we place an additional buffer that favors keeping in more out-of-range sequences than removing in-range sequences; basically, we update the threshold such that we search for $d_k > b + 0.2$. Third, instead of traversing the list in order and finding the *first* sequence talfre hat satisfies $d_k > b + 0.2$, we perform a binary search through the list, which makes it much less likely to stop at a very low positioned sequenced in the list.

As such, if the number of sequences is higher than 2,000, we perform a binary search through the list to find an (approximate) "upper bound" above which all sequences (likely) have a distance value larger than the maximum allowed threshold b (+0.2). This operation is quick, requiring $\log(n)$ Kimura computations on average (where n is the number of hits). In other words, for a list of 100,000 hits, we require (on average) 16.6 Kimura computations, where each computation is a simple run through an *existing* local alignment, counting up the transition and transversion changes required in (Equation 4.8).

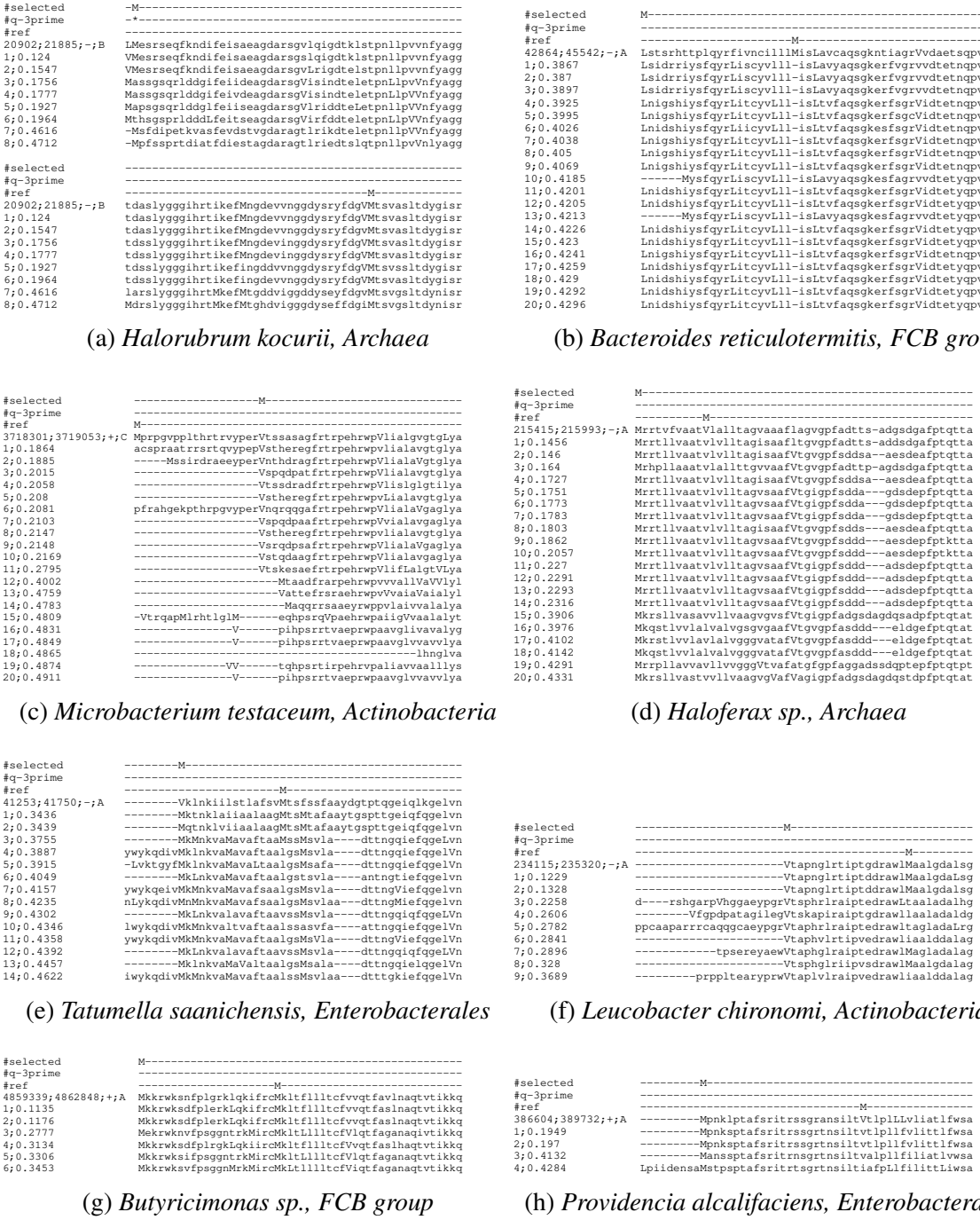
We effectively now have a list of sequences whose Kimura is in the range $[0, b + 0.2]$. We don't perform the same operation to find the lower limit because our desired threshold of 0.1 is close to 0, leaving us a small buffer region. Instead, we shuffle the list and go through it one sequence at a time until we have 50 sequences that satisfy our requirements. In the worst case, we would have to go through all sequences in this range, especially if the target sequences are very similar to the query or each other.

B.5.0.4 Optimality in the face of randomness

If we had gone through every sequence in the list, computed d_k, sorted it, then filtered out the sequences outside the range $[a, b]$, we would still have to randomly select only N (from the possibly thousands) of the remaining sequences. This reduction, which comes from the requirement of the subsequent multiple sequence alignment step, means that even if we were to act optimally at the initial stages, the act of random selection reduces the likelihood that our final set is actually optimal. In other words, we can act sub-optimally

(as suggested in the above heuristics) at the initial steps without necessarily observing a change in the final outcome. In practice, we did not see a tangible difference in StartLink's final performance by acting optimally or sub-optimally at these initial stages.

B.6 Supplementary Figures



(a) *Halorubrum kocurii, Archaea* (b) *Bacteroides reticulotermitis, FCB group*

(c) *Microbacterium testaceum, Actinobacteria* (d) *Haloferax sp., Archaea*

(e) *Tatumella saanichensis, Enterobacterales* (f) *Leucobacter chironomi, Actinobacteria*

(g) *Butyricimonas sp., FCB group* (h) *Providencia alcalifaciens, Enterobacterales*

Figure B.8: Examples of multiple sequence alignments that show a mismatch between StartLink+ (#selected) and PGAP's prediction (#ref). The examples are drawn from 8 genomes coming from 4 clades: Actinobacteria, Archaea, Enterobacterales, and FCB group. The "M" in #selected and #ref shows the positions of the predicted start, the "*" in #q-3prime shows the position of the 3' end of the upstream gene (if it exists). Following #ref is the query sequence, followed by the target sequences. Each target sequence has a floating point number representing the Kimura distance to the query.

APPENDIX C

METAGENEMARKS

C.1 Building average motif and spacer models

C.1.1 Clustering models

The difficulty in merging motif models, represented as positional Markov models, is that the significance of a position can change between models. For example, consider AAGGAG, AGGAGG, AGGAGA, the consensus sequences of three motif models each of width 6. A direct merging off these models results in conflicts because AAGGAG seems to be ”shifted” relative to the other two. Instead, we merge AGGAGG and AGGAGA together, and leave AAGGAG as its own model.

More formally, for a given set of consensus sequences, we find a small number of clusters such that, within each cluster, pairs of consensus sequences differ by at most one nucleotide. This allows us to merge ”similar” motif models with each other, reducing the overall number of motif models while still ensuring that the probability values per position reflect a stable average.

The algorithm can be described as a greedy, single-pass clustering algorithm and works as follows: We start with an empty set of clusters. Given a list of motif model consensus sequences, sort them by how frequently they occur. For each sequence s in that order, we loop over the current set of clusters, and find the first cluster for which all members differ from s by at most one nucleotide. If such a cluster is found, add s to it and move to the next sequence. Otherwise, create a new cluster and add s to it.

Note that this approach encourages the most frequently occurring motifs to define the creation of clusters (since they are analyzed earlier). This is preferred as it accounts for the density of datapoints, which is typically accounted for when performing standard clustering

in a Euclidean space. A manual inspection of all created models shows that the number of clusters is equal to the minimum possible number of clusters that satisfy the condition of a pairwise edit-distance less than or equal to 1, per cluster (although this is not a guaranteed outcome of this algorithm, in general).

C.2 Merging spacer distributions

A key element of the motif models is the distance of the motif from the start codon. For example, the distances between ribosomal binding sites and gene starts is non-uniformly distributed, with bell-like curves that have a mean around 4-8 nucleotides from the gene start.

What is less understood, however, are the factors that shift this mean from one species to another. For example, Figure C.1 shows the distribution of the peaks of spacer distributions for two types of RBS models derived by GeneMarkS-2, for genomes whose GC content lies in the range [45,50). We first note that even for models with the AGGAGG consensus, the peaks of spacer distributions can vary significantly from 4 to 8 nucleotides from the gene-start. Second, the average peak position of the AAGGAG consensus is 6 while that of AGGAGG is 7. This is interesting because at first glance, one would assume that compared to AGGAGG, AAGGAG would on average align to the 16S rRNA tail at one position further from the gene-start.

One approach to merging spacer distributions is to only merge those that have the same peak location. However, as shown in Figure C.1, this can generate many spacer distributions per consensus. That means that for a given GC bin with multiple unique consensus sequences, the effective number of models can grow significantly. For example, for RBS models from group A genomes in the range [40, 45], we have four unique consensus sequences (clustered into two groups). If we consider the spacer models separately, we will have around 20 separate combinations of motif/spacer models. We can reduce this to just 2 by averaging all spacer distributions within a cluster, as shown in Figure 5.4.

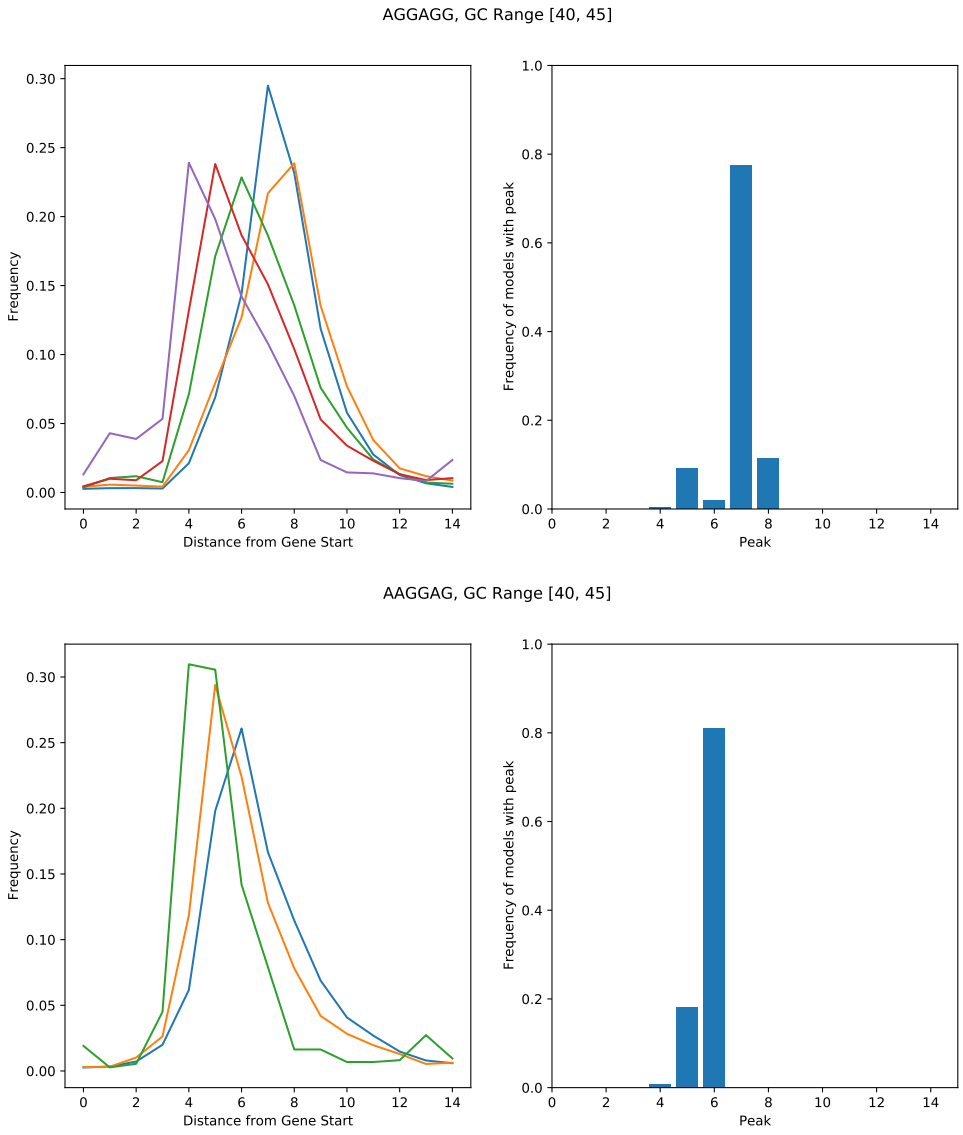


Figure C.1: Average spacer distributions per peak (left) and the frequency of spacer peak positions (right) for RBS models with AGGAGG (top) and AAGGAG (bottom) consensus sequences. This is computed over the set of representative bacterial group A genomes in the GC range [45, 50).

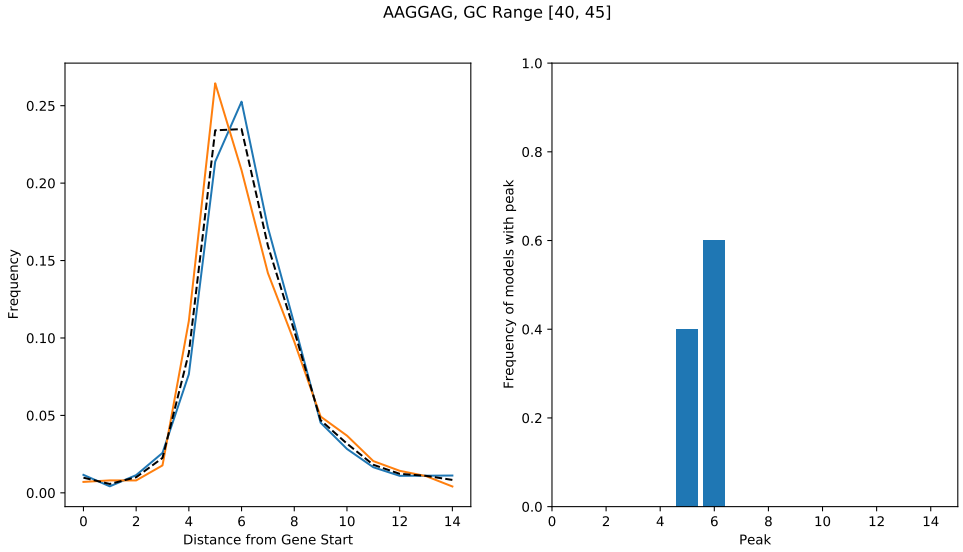


Figure C.2: The per-peak average spacer distributions and the total average (dashed) of AAGGAG consensus sequences from group A genomes in the GC content range of [40,45).

For example, the average of the spacer distributions for consensus AAGGAG is shown in Figure C.2. Note that if we were to use the individual spacer models, their scores would be weighted by the prior probabilities of a given peak. The average spacer accounts for the frequency of different peaks implicitly, and therefore inherently penalizes rarely occurring spacer models. This allows us to achieve behavior similar to having individual spacer models, but without the need to store multiple spacer models.

C.3 Automatic identification of genetic code

MetaGeneMarkS and MetaProdigal are both able to automatically detect a genome's genetic code (4 or 11). This section will describe the method used by MetaGeneMarkS, and how it compares in performance to MetaProdigal.[1]

[1]This part of the project was primarily lead by Dr. Alex Lomsadze and was later included into MetaGeneMarkS

C.3.1 Method

MetaGeneMarkS has two sets of the model files: one for species with genetic code 11 and another for species with genetic code 4. In comparison with the more frequent genetic code 11, the models for genetic code 4 have differences in modeling of the "TGA" codon as well as the gene start signals. All the other model parameters are identical (CITE: Besemer and Borodovsky 1999). To determine the correct model for gene prediction, MetaGeneMarkS runs each of the two sets of models and computes the log-odds ratio L of the two likelihoods

$$Prior\,(\text{code } i)\max\{P(\text{sequence} \parallel \text{model code } i)\} \qquad i = 4, 11 \tag{C.1}$$

If L is greater than 1, MetaGeneMarkS uses code 11; otherwise it uses code 4. The prior probabilities were optimized on the training set of fragments from 12 randomly selected genomes with known genetic codes (6 genomes for each genetic code).

C.3.2 Testing and accuracy of genetic code detection

For testing, we downloaded the representative RefSeq genomes of 405 archaea and 11,265 bacteria[2] and 18 genomes of *Hodgkinia* with genetic code 4. All species in this set except the 167 bacteria were annotated as using genetic code 11.

In MetaGeneMarkS and MetaProdigal, the genetic code is identified on a gene by gene basis. We therefore measure the accuracy of genetic code detection as follows: a genome's genetic code is determined by the majority of genetic code predictions for its genes. In other words, if predictions for more than 50% of genes match the correct genetic code, we label this as true positive; otherwise, it is counted as an error. We compare genetic codes identified by MetaGeneMarkS and MetaProdigal with the annotated genetic codes of the 11,688 genomes.

Genetic code was correctly detected by MetaGeneMarkS in all 167 genomes with ge-

[2]Link: https://www.ncbi.nlm.nih.gov/genome/browse#!/prokaryotes/

Table C.1: False negative rate by MetaGeneMarkS and MetaProdigal on 15 genomes with a large difference in genetic code predictions by MetaGeneMarkS and MetaProdigal. The reference set only includes genes from RefSeq annotation that are supported by homologous proteins.

Species names	Number of PGAP genes supported by homology	% of genes missed by MetaGeneMarkS	% of genes missed by MetaProdigal
Acholeplasma axanthum strain NCTC10138	1,389	**1.2**	17.1
Acholeplasma hippikon strain NCTC1072	1,121	**1.0**	18.4
Acholeplasma modicum ATCC 29102	956	**1.1**	17.4
Anaerococcus obesiensis ph10	1,914	**4.0**	18.5
Anaerococcus octavius strain NCTC9810	1,602	**1.7**	17.1
Anaerococcus vaginali ATCC 51170	1,714	**2.0**	16.0
Helicobacter bizzozeronii CIII-1	1,613	**6.7**	32.9
Peptoniphilus duerdenii ATCC BAA-1640	1,789	**2.2**	23.5
Peptoniphilus grossensis ph5	1,845	**1.3**	21.1
Peptoniphilus lacydonensis strain EL1	1,648	**2.5**	23.5
Peptoniphilus phoceensis strain SIT15	1,562	**2.2**	21.1
Peptoniphilus raoultii strain KHD4	1,461	**3.3**	20.1
Peptoniphilus senegalensis JC140	1,645	**2.5**	22.8
Peptoniphilus timonensis JC401	1,533	**5.2**	25.8
Sneathia mania strain SN35	1,084	**1.7**	19.2

netic code 4 and in all 11,521 genomes with genetic code 11. MetaProdigal incorrectly assigned genetic code 11 to all 18 *Hodgkinia* genomes; furthermore 51 genomes with genetic code 11 were incorrectly assigned to genetic code 4. While [69] claimed that errors in MetaProdigal's genetic code assignment do not affect its gene-level performance, this is due to their limited set of genetic code 4 genomes (basically limited to *Mycoplasmas* and similar species). Our tests on *Hodgkinia* show that errors in genetic code assignment do in fact significantly alter gene-level performance, sometimes by up to 20% (Table C.1).

Particularly, MetaProdigal predicted genetic code 4 for 138,437 genes and genetic code 11 for 3,560 genes. On the other hand, MetaGeneMarkS incorrectly predicted genetic codes 4 and 11 for 1,471 and 1,982 genes, respectively.

MetaProdigal uses a set of 50 species specific parameters for gene prediction in metagenomes. Four out of the 50 MetaProdigal models are from species of *Mycoplasma*. In the set of

167 genomes with genetic code 4, 93 are *Mycoplasmas*. MetaProdigal correctly detects genetic code 4 in all 93 *Mycoplasma* genomes, slightly outperforming MetaGeneMarkS on the gene level. However, the genomes from *Candidatus Hodgkinia cicadicola*, which are evolutionarily far from *Mycoplasmas* but use genetic code 4, were all misclassified by MetaProdigal. This is likely because these genomes were not used in training of MetaProdigal, which shows some overfitting to *Mycoplasmas*.. Genome GC of Hodgkinia varies in the range 38-58%, which is higher than that of Mycoplasma, 22-40%. The genetic code and genes in these genomes were correctly predicted by MetaGeneMarkS.

Other incorrect genetic code predictions by MetaProdigal were also observed when the coding hexamer frequencies in genetic code 11 genomes are similar to *Mycoplasma*. Such incorrect predictions were observed for the 3 genomes of *Acholeplasma*, the 3 genomes of *Anaerococcus*, and the 7 genomes of *Peptoniphilus* species Table C.1.

APPENDIX D

PHYLOGENETIC DISTRIBUTION OF GENOMES INTO GROUPS

The following trees show the phylogenetic distribution of archaeal and bacterial genomes into groups A, B, C, D, and X, as defined in Chapter 3.

Group A

a: Total number of genomes in the taxon
b: Number of group A genomes in the taxon
c: Percentage of group A genomes in the taxon

	a	b	c
cellular organisms	5007	2974	59.4
|__ Bacteria	4769	2935	61.5
|__ Proteobacteria	1854	1570	84.7
|__ Gammaproteobacteria	711	631	88.7
|__ Enterobacterales	112	97	86.6
|__ Enterobacteriaceae	41	32	78.0
|__ unclassified Enterobacteriaceae	13	7	53.8
|__ Erwiniaceae	30	28	93.3
|__ Erwinia	10	10	100.0
|__ Buchnera	8	8	100.0
|__ Buchnera aphidicola	8	8	100.0
|__ Pantoea	8	8	100.0
|__ Yersiniaceae	14	14	100.0
|__ Serratia	8	8	100.0
|__ Morganellaceae	13	10	76.9
|__ Pectobacteriaceae	8	8	100.0
|__ Alteromonadales	94	94	100.0
|__ Alteromonadaceae	34	34	100.0
|__ Marinobacter	12	12	100.0
|__ Shewanellaceae	17	17	100.0
|__ Shewanella	17	17	100.0
|__ Pseudoalteromonadaceae	16	16	100.0
|__ Pseudoalteromonas	15	15	100.0
|__ Idiomarinaceae	9	9	100.0
|__ Idiomarina	9	9	100.0
|__ Colwelliaceae	6	6	100.0
|__ Pseudomonadales	81	81	100.0

```
|     |     |     |     |__ Moraxellaceae                                        41      41      100.0
|     |     |     |     |     |__ Acinetobacter                                25      25      100.0
|     |     |     |     |     |__ Psychrobacter                                8       8       100.0
|     |     |     |     |     |__ Moraxella                                    6       6       100.0
|     |     |     |     |__ Pseudomonadaceae                                     40      40      100.0
|     |     |     |     |     |__ Pseudomonas                                  38      38      100.0
|     |     |     |__ Oceanospirillales                                          73      72      98.6
|     |     |     |     |__ Oceanospirillaceae                                   28      28      100.0
|     |     |     |     |     |__ Marinomonas                                  7       7       100.0
|     |     |     |     |__ Halomonadaceae                                       28      27      96.4
|     |     |     |     |     |__ Halomonas                                    19      19      100.0
|     |     |     |     |__ Alcanivoracaceae                                     6       6       100.0
|     |     |     |     |     |__ Alcanivorax                                  6       6       100.0
|     |     |     |     |__ Hahellaceae                                          6       6       100.0
|     |     |     |__ Vibrionales                                                49      48      98.0
|     |     |     |     |__ Vibrionaceae                                         49      48      98.0
|     |     |     |     |     |__ Vibrio                                       27      26      96.3
|     |     |     |     |     |__ Photobacterium                               13      13      100.0
|     |     |     |__ Chromatiales                                               42      40      95.2
|     |     |     |     |__ Ectothiorhodospiraceae                               21      20      95.2
|     |     |     |     |     |__ Thioalkalivibrio                             11      11      100.0
|     |     |     |     |__ Chromatiaceae                                        18      17      94.4
|     |     |     |__ Xanthomonadales                                            62      35      56.5
|     |     |     |     |__ Xanthomonadaceae                                     48      22      45.8
|     |     |     |     |     |__ Stenotrophomonas                             10      8       80.0
|     |     |     |     |__ Rhodanobacteraceae                                   14      13      92.9
|     |     |     |__ Thiotrichales                                              34      31      91.2
|     |     |     |     |__ Piscirickettsiaceae                                  21      20      95.2
|     |     |     |     |     |__ Thiomicrospira                               10      10      100.0
|     |     |     |     |__ Thiotrichaceae                                       7       6       85.7
|     |     |     |__ Cellvibrionales                                            31      31      100.0
|     |     |     |     |__ Cellvibrionaceae                                     13      13      100.0
|     |     |     |     |__ Halieaceae                                           8       8       100.0
|     |     |     |     |__ Spongiibacteraceae                                   6       6       100.0
|     |     |     |__ Legionellales                                              38      31      81.6
|     |     |     |     |__ Legionellaceae                                       32      28      87.5
|     |     |     |     |     |__ Legionella                                   30      26      86.7
|     |     |     |__ unclassified Gammaproteobacteria                           26      18      69.2
|     |     |     |     |__ unclassified Gammaproteobacteria (miscellaneous)     7       6       85.7
|     |     |     |__ Pasteurellales                                             19      16      84.2
|     |     |     |     |__ Pasteurellaceae                                      19      16      84.2
|     |     |     |__ Aeromonadales                                              15      14      93.3
|     |     |     |     |__ Aeromonadaceae                                       9       9       100.0
```

```
|    |    |    |    |    |__ Aeromonas                              7      7      100.0
|    |    |    |__ Methylococcales                                  20     11     55.0
|    |    |    |    |__ Methylococcaceae                            19     10     52.6
|    |    |    |__ Nevskiales                                       7      6      85.7
|    |    |__ Alphaproteobacteria                                   667    583    87.4
|    |    |    |__ Rhizobiales                                      225    221    98.2
|    |    |    |    |__ Bradyrhizobiaceae                           44     43     97.7
|    |    |    |    |    |__ Bradyrhizobium                         18     18     100.0
|    |    |    |    |    |__ Bosea                                  8      8      100.0
|    |    |    |    |    |__ Rhodopseudomonas                       8      7      87.5
|    |    |    |    |    |__ Afipia                                 6      6      100.0
|    |    |    |    |__ Rhizobiaceae                                41     40     97.6
|    |    |    |    |    |__ Rhizobium/Agrobacterium group          29     29     100.0
|    |    |    |    |    |    |__ Rhizobium                         19     19     100.0
|    |    |    |    |    |    |__ Agrobacterium                     8      8      100.0
|    |    |    |    |__ Phyllobacteriaceae                          28     27     96.4
|    |    |    |    |    |__ Mesorhizobium                          11     11     100.0
|    |    |    |    |__ Bartonellaceae                              23     23     100.0
|    |    |    |    |    |__ Bartonella                             23     23     100.0
|    |    |    |    |__ Hyphomicrobiaceae                           23     23     100.0
|    |    |    |    |    |__ Devosia                                11     11     100.0
|    |    |    |    |__ Methylobacteriaceae                         19     18     94.7
|    |    |    |    |    |__ Methylobacterium                       14     13     92.9
|    |    |    |    |__ Aurantimonadaceae                           11     11     100.0
|    |    |    |    |__ Methylocystaceae                            10     10     100.0
|    |    |    |    |__ Beijerinckiaceae                            7      7      100.0
|    |    |    |    |__ Xanthobacteraceae                           7      7      100.0
|    |    |    |    |__ Rhodobiaceae                                6      6      100.0
|    |    |    |__ Rhodobacterales                                  166    161    97.0
|    |    |    |    |__ Rhodobacteraceae                            145    140    96.6
|    |    |    |    |    |__ Paracoccus                             12     12     100.0
|    |    |    |    |    |__ Sulfitobacter                          8      8      100.0
|    |    |    |    |    |__ Loktanella                             6      6      100.0
|    |    |    |    |    |__ Roseobacter                            6      6      100.0
|    |    |    |    |__ Hyphomonadaceae                             20     20     100.0
|    |    |    |    |    |__ Hyphomonas                             9      9      100.0
|    |    |    |__ Sphingomonadales                                 105    91     86.7
|    |    |    |    |__ Sphingomonadaceae                           83     72     86.7
|    |    |    |    |    |__ Sphingomonas                           34     24     70.6
|    |    |    |    |    |__ Novosphingobium                        17     17     100.0
|    |    |    |    |    |__ Sphingobium                            13     13     100.0
|    |    |    |    |    |__ Sphingopyxis                           9      9      100.0
|    |    |    |    |__ Erythrobacteraceae                          22     19     86.4
```

```
|    |    |    |    |    |__ Erythrobacter                                          12      10      83.3
|    |    |    |__ Rhodospirillales                                                 72      64      88.9
|    |    |    |    |__ Rhodospirillaceae                                           32      31      96.9
|    |    |    |    |__ Acetobacteraceae                                            37      31      83.8
|    |    |    |    |    |__ Acetobacter                                            10      8       80.0
|    |    |    |__ Caulobacterales                                                  23      20      87.0
|    |    |    |    |__ Caulobacteraceae                                            23      20      87.0
|    |    |    |    |    |__ Brevundimonas                                          10      7       70.0
|    |    |    |__ unclassified Alphaproteobacteria                                 13      12      92.3
|    |    |    |    |__ unclassified Alphaproteobacteria (miscellaneous)            7       6       85.7
|    |    |__ Betaproteobacteria                                                    273     177     64.8
|    |    |    |__ Burkholderiales                                                  166     88      53.0
|    |    |    |    |__ Oxalobacteraceae                                            30      26      86.7
|    |    |    |    |    |__ Massilia                                               8       8       100.0
|    |    |    |    |    |__ Herbaspirillum                                         7       7       100.0
|    |    |    |    |__ Burkholderiaceae                                            34      25      73.5
|    |    |    |    |    |__ Paraburkholderia                                       8       7       87.5
|    |    |    |    |__ Alcaligenaceae                                              23      20      87.0
|    |    |    |    |__ Comamonadaceae                                              50      10      20.0
|    |    |    |__ Neisseriales                                                     42      42      100.0
|    |    |    |    |__ Neisseriaceae                                               23      23      100.0
|    |    |    |    |    |__ Neisseria                                              10      10      100.0
|    |    |    |    |__ Chromobacteriaceae                                          19      19      100.0
|    |    |    |__ Rhodocyclales                                                    19      19      100.0
|    |    |    |    |__ Rhodocyclaceae                                              19      19      100.0
|    |    |    |__ Methylophilales                                                  14      10      71.4
|    |    |    |    |__ Methylophilaceae                                            12      10      83.3
|    |    |__ delta/epsilon subdivisions                                            196     173     88.3
|    |    |    |__ Deltaproteobacteria                                              123     110     89.4
|    |    |    |    |__ Desulfovibrionales                                          44      44      100.0
|    |    |    |    |    |__ Desulfovibrionaceae                                    35      35      100.0
|    |    |    |    |    |    |__ Desulfovibrio                                     31      31      100.0
|    |    |    |    |__ Desulfobacterales                                           22      22      100.0
|    |    |    |    |    |__ Desulfobacteraceae                                     15      15      100.0
|    |    |    |    |    |__ Desulfobulbaceae                                       7       7       100.0
|    |    |    |    |__ Desulfuromonadales                                          16      16      100.0
|    |    |    |    |    |__ Geobacteraceae                                         10      10      100.0
|    |    |    |    |    |    |__ Geobacter                                         7       7       100.0
|    |    |    |    |    |__ Desulfuromonadaceae                                    6       6       100.0
|    |    |    |    |__ Myxococcales                                                15      11      73.3
|    |    |    |    |    |__ Cystobacterineae                                       11      10      90.9
|    |    |    |__ Epsilonproteobacteria                                            73      63      86.3
|    |    |    |    |__ Campylobacterales                                           67      58      86.6
```

```
|    |    |    |    |    |__ Helicobacteraceae                  34      30      88.2
|    |    |    |    |    |    |__ Helicobacter                  28      27      96.4
|    |    |    |    |    |__ Campylobacteraceae                 32      28      87.5
|    |    |    |    |    |    |__ Campylobacter                 19      17      89.5
|    |    |    |    |    |    |__ Arcobacter                    7       6       85.7
|    |__ Terrabacteria group                                    2228    1206    54.1
|    |    |__ Firmicutes                                        1064    1028    96.6
|    |    |    |__ Bacilli                                      597     573     96.0
|    |    |    |    |__ Bacillales                              342     342     100.0
|    |    |    |    |    |__ Bacillaceae                        165     165     100.0
|    |    |    |    |    |    |__ Bacillus                      97      97      100.0
|    |    |    |    |    |    |__ Lysinibacillus                8       8       100.0
|    |    |    |    |    |    |__ Oceanobacillus                6       6       100.0
|    |    |    |    |    |__ Paenibacillaceae                   71      71      100.0
|    |    |    |    |    |    |__ Paenibacillus                 50      50      100.0
|    |    |    |    |    |    |__ Brevibacillus                 8       8       100.0
|    |    |    |    |    |__ Staphylococcaceae                  38      38      100.0
|    |    |    |    |    |    |__ Staphylococcus                28      28      100.0
|    |    |    |    |    |__ Planococcaceae                     24      24      100.0
|    |    |    |    |    |__ Listeriaceae                       11      11      100.0
|    |    |    |    |    |    |__ Listeria                      9       9       100.0
|    |    |    |    |    |__ Alicyclobacillaceae                11      11      100.0
|    |    |    |    |    |    |__ Alicyclobacillus              8       8       100.0
|    |    |    |    |    |__ Bacillales incertae sedis          9       9       100.0
|    |    |    |    |    |__ Thermoactinomycetaceae             6       6       100.0
|    |    |    |    |__ Lactobacillales                         255     231     90.6
|    |    |    |    |    |__ Lactobacillaceae                   141     140     99.3
|    |    |    |    |    |    |__ Lactobacillus                 131     130     99.2
|    |    |    |    |    |    |__ Pediococcus                   9       9       100.0
|    |    |    |    |    |__ Enterococcaceae                    23      23      100.0
|    |    |    |    |    |    |__ Enterococcus                  17      17      100.0
|    |    |    |    |    |__ Streptococcaceae                   41      22      53.7
|    |    |    |    |    |    |__ Streptococcus                 39      21      53.8
|    |    |    |    |    |__ Leuconostocaceae                   23      19      82.6
|    |    |    |    |    |    |__ Weissella                     10      10      100.0
|    |    |    |    |    |    |__ Leuconostoc                   8       6       75.0
|    |    |    |    |    |__ Carnobacteriaceae                  19      19      100.0
|    |    |    |    |    |    |__ Carnobacterium                10      10      100.0
|    |    |    |    |    |__ Aerococcaceae                      8       8       100.0
|    |    |    |__ Clostridia                                   370     366     98.9
|    |    |    |    |__ Clostridiales                           335     334     99.7
|    |    |    |    |    |__ Lachnospiraceae                    101     101     100.0
|    |    |    |    |    |    |__ unclassified Lachnospiraceae  33      33      100.0
```

```
|    |    |    |    |    |    |__ Lachnoclostridium                                   14     14     100.0
|    |    |    |    |    |    |__ Butyrivibrio                                        10     10     100.0
|    |    |    |    |    |    |__ Blautia                                             9      9      100.0
|    |    |    |    |    |__ Clostridiaceae                                          82     82     100.0
|    |    |    |    |    |    |__ Clostridium                                         57     57     100.0
|    |    |    |    |    |__ Ruminococcaceae                                         45     45     100.0
|    |    |    |    |    |    |__ Ruminococcus                                        14     14     100.0
|    |    |    |    |    |    |__ Ruminiclostridium                                   13     13     100.0
|    |    |    |    |    |__ Peptococcaceae                                          28     27     96.4
|    |    |    |    |    |    |__ Desulfotomaculum                                    9      9      100.0
|    |    |    |    |    |    |__ Desulfosporosinus                                   7      7      100.0
|    |    |    |    |    |__ Peptostreptococcaceae                                   19     19     100.0
|    |    |    |    |    |__ unclassified Clostridiales                              18     18     100.0
|    |    |    |    |    |    |__ unclassified Clostridiales (miscellaneous)          6      6      100.0
|    |    |    |    |    |__ Eubacteriaceae                                          17     17     100.0
|    |    |    |    |    |    |__ Eubacterium                                         12     12     100.0
|    |    |    |    |    |__ Clostridiales incertae sedis                            10     10     100.0
|    |    |    |    |    |    |__ Clostridiales Family XIII. Incertae Sedis           7      7      100.0
|    |    |    |    |__ Thermoanaerobacterales                                      24     21     87.5
|    |    |    |    |    |__ Thermoanaerobacteraceae                                 13     13     100.0
|    |    |    |    |    |__ Thermoanaerobacterales Family III. Incertae Sedis       7      7      100.0
|    |    |    |    |__ Halanaerobiales                                             8      8      100.0
|    |    |    |__ Negativicutes                                                   41     41     100.0
|    |    |    |    |__ Selenomonadales                                             21     21     100.0
|    |    |    |    |    |__ Selenomonadaceae                                        15     15     100.0
|    |    |    |    |    |    |__ Selenomonas                                         8      8      100.0
|    |    |    |    |    |__ Sporomusaceae                                           6      6      100.0
|    |    |    |    |__ Veillonellales                                              16     16     100.0
|    |    |    |    |    |__ Veillonellaceae                                         16     16     100.0
|    |    |    |    |    |    |__ Megasphaera                                         7      7      100.0
|    |    |    |__ Tissierellia                                                    32     29     90.6
|    |    |    |    |__ Tissierellales                                              26     23     88.5
|    |    |    |    |    |__ Peptoniphilaceae                                        26     23     88.5
|    |    |    |    |    |    |__ Peptoniphilus                                       13     10     76.9
|    |    |    |    |    |    |__ Anaerococcus                                        8      8      100.0
|    |    |    |    |__ unclassified Tissierellia                                   6      6      100.0
|    |    |    |__ Erysipelotrichia                                                21     17     81.0
|    |    |    |    |__ Erysipelotrichales                                          21     17     81.0
|    |    |    |    |    |__ Erysipelotrichaceae                                     21     17     81.0
|    |    |__ Actinobacteria                                                       859    78     9.1
|    |    |    |__ Actinobacteria                                                  807    38     4.7
|    |    |    |    |__ Bifidobacteriales                                           36     16     44.4
|    |    |    |    |    |__ Bifidobacteriaceae                                      36     16     44.4
```

```
|     |     |     |     |     |     |__ Bifidobacterium                  30     16     53.3
|     |     |     |     |__ Micrococcales                                211    10     4.7
|     |     |     |     |     |__ Micrococcaceae                         51     8      15.7
|     |     |     |__ Coriobacteriia                                     34     34     100.0
|     |     |     |     |__ Coriobacteriales                             23     23     100.0
|     |     |     |     |     |__ Atopobiaceae                           12     12     100.0
|     |     |     |     |     |     |__ Atopobium                        7      7      100.0
|     |     |     |     |     |__ Coriobacteriaceae                      11     11     100.0
|     |     |     |     |__ Eggerthellales                               11     11     100.0
|     |     |     |     |     |__ Eggerthellaceae                        11     11     100.0
|     |     |__ Tenericutes                                              116    72     62.1
|     |     |     |__ Mollicutes                                         115    71     61.7
|     |     |     |     |__ Mycoplasmatales                              73     33     45.2
|     |     |     |     |     |__ Mycoplasmataceae                       73     33     45.2
|     |     |     |     |     |     |__ Mycoplasma                       70     30     42.9
|     |     |     |     |__ Entomoplasmatales                            22     19     86.4
|     |     |     |     |     |__ Spiroplasmataceae                      15     12     80.0
|     |     |     |     |     |     |__ Spiroplasma                      15     12     80.0
|     |     |     |     |     |__ Entomoplasmataceae                     7      7      100.0
|     |     |     |     |__ Acholeplasmatales                            20     19     95.0
|     |     |     |     |     |__ Acholeplasmataceae                     20     19     95.0
|     |     |     |     |     |     |__ Acholeplasma                     10     10     100.0
|     |     |     |     |     |     |__ Candidatus Phytoplasma           10     9      90.0
|     |     |__ Chloroflexi                                              21     21     100.0
|     |     |     |__ Chloroflexia                                       6      6      100.0
|     |__ PVC group                                                      52     28     53.8
|     |     |__ Chlamydiae                                               17     15     88.2
|     |     |     |__ Chlamydiia                                         17     15     88.2
|     |     |     |     |__ Parachlamydiales                             8      8      100.0
|     |     |     |     |__ Chlamydiales                                 9      7      77.8
|     |     |     |     |     |__ Chlamydiaceae                          9      7      77.8
|     |     |     |     |     |     |__ Chlamydia/Chlamydophila group    9      7      77.8
|     |     |     |     |     |     |     |__ Chlamydia                  9      7      77.8
|     |     |__ Planctomycetes                                           16     11     68.8
|     |     |     |__ Planctomycetia                                     15     11     73.3
|     |     |     |     |__ Planctomycetales                             13     9      69.2
|     |__ Spirochaetes                                                   60     26     43.3
|     |     |__ Spirochaetia                                             60     26     43.3
|     |     |     |__ Spirochaetales                                     40     16     40.0
|     |     |     |     |__ Borreliaceae                                 12     9      75.0
|     |     |     |     |__ Spirochaetaceae                              28     7      25.0
|     |     |     |__ Leptospirales                                      16     6      37.5
|     |     |     |     |__ Leptospiraceae                               16     6      37.5
```

| | | | | |__ Leptospira	14	6	42.9
| |__ Thermotogae	19	19	100.0
| | |__ Thermotogae	19	19	100.0
| | | |__ Thermotogales	12	12	100.0
| | | | |__ Thermotogaceae	7	7	100.0
| |__ Fusobacteria	19	18	94.7
| | |__ Fusobacteriia	19	18	94.7
| | | |__ Fusobacteriales	19	18	94.7
| | | | |__ Fusobacteriaceae	11	11	100.0
| | | | | |__ Fusobacterium	8	8	100.0
| | | | |__ Leptotrichiaceae	8	7	87.5
| |__ Acidobacteria	24	17	70.8
| | |__ Acidobacteriia	17	15	88.2
| | | |__ Acidobacteriales	17	15	88.2
| | | | |__ Acidobacteriaceae	17	15	88.2
| |__ Synergistetes	13	13	100.0
| | |__ Synergistia	13	13	100.0
| | | |__ Synergistales	13	13	100.0
| | | | |__ Synergistaceae	13	13	100.0
| |__ Aquificae	14	11	78.6
| | |__ Aquificae	14	11	78.6
| | | |__ Aquificales	11	8	72.7
| |__ Nitrospirae	6	6	100.0
| | |__ Nitrospira	6	6	100.0
| | | |__ Nitrospirales	6	6	100.0
| | | | |__ Nitrospiraceae	6	6	100.0
| |__ Thermodesulfobacteria	6	5	83.3
| |__ unclassified Bacteria	5	4	80.0
| |__ FCB group	455	3	0.7
| |__ Deferribacteres	6	3	50.0
| |__ Chrysiogenetes	2	2	100.0
| |__ Dictyoglomi	1	1	100.0
| |__ Calditrichaeota	1	1	100.0
| |__ Caldiserica	1	1	100.0
| |__ Nitrospinae/Tectomicrobia group	1	1	100.0
|__ Archaea	238	39	16.4
| |__ Euryarchaeota	190	27	14.2
| | |__ Methanobacteria	18	13	72.2
| | | |__ Methanobacteriales	18	13	72.2
| | | | |__ Methanobacteriaceae	17	12	70.6
| | | | | |__ Methanobacterium	7	6	85.7
| | |__ Methanococci	15	12	80.0
| | | |__ Methanococcales	15	12	80.0

```
|     |     |     |     |__ Methanococcaceae                                7        7        100.0
|     |__ TACK group                                                      47       12       25.5
|     |     |__ Crenarchaeota                                             35       12       34.3
|     |     |     |__ Thermoprotei                                        35       12       34.3
|     |     |     |     |__ Desulfurococcales                             13       12       92.3
|     |     |     |     |     |__ Desulfurococcaceae                      9        9        100.0
```

Group B

a: Total number of genomes in the taxon
b: Number of group B genomes in the taxon
c: Percentage of group B genomes in the taxon

	a	b	c
cellular organisms	5007	495	9.9
|__ Bacteria	4769	495	10.4
| |__ FCB group	455	409	89.9
| | |__ Bacteroidetes/Chlorobi group	450	408	90.7
| | | |__ Bacteroidetes	436	400	91.7
| | | | |__ Flavobacteriia	190	187	98.4
| | | | | |__ Flavobacteriales	188	185	98.4
| | | | | | |__ Flavobacteriaceae	174	172	98.9
| | | | | | | |__ Flavobacterium	30	30	100.0
| | | | | | | |__ Chryseobacterium	24	24	100.0
| | | | | | | |__ Capnocytophaga	8	8	100.0
| | | | | | | |__ Lacinutrix	5	5	100.0
| | | | | | | |__ Aquimarina	5	5	100.0
| | | | | | | |__ Psychroserpens	5	5	100.0
| | | | | | | |__ Maribacter	4	4	100.0
| | | | | | | |__ Polaribacter	4	4	100.0
| | | | | | | |__ Nonlabens	5	4	80.0
| | | | | | | |__ Cellulophaga	4	4	100.0
| | | | | | | |__ Tenacibaculum	4	4	100.0
| | | | | | | |__ Arenibacter	3	3	100.0
| | | | | | | |__ Mangrovimonas	3	3	100.0
| | | | | | | |__ Leeuwenhoekiella	3	3	100.0
| | | | | | | |__ Muricauda	3	3	100.0
| | | | | | | |__ Kordia	3	3	100.0
| | | | | | | |__ Gillisia	3	3	100.0
| | | | | | | |__ Psychroflexus	3	3	100.0
| | | | | | |__ Blattabacteriaceae	8	7	87.5
| | | | | | | |__ Blattabacterium	8	7	87.5
| | | | |__ Bacteroidia	137	130	94.9
| | | | | |__ Bacteroidales	127	120	94.5
| | | | | | |__ Prevotellaceae	52	51	98.1
| | | | | | | |__ Prevotella	49	49	100.0
| | | | | | |__ Porphyromonadaceae	33	31	93.9
| | | | | | | |__ Porphyromonas	15	13	86.7
| | | | | | | |__ Dysgonomonas	5	5	100.0

```
|    |    |    |    |    |    |    |__ Parabacteroides                  3       3       100.0
|    |    |    |    |    |    |__ Bacteroidaceae                       23      21      91.3
|    |    |    |    |    |    |    |__ Bacteroides                      23      21      91.3
|    |    |    |    |    |    |__ Rikenellaceae                        12      11      91.7
|    |    |    |    |    |    |    |__ Alistipes                        10      10      100.0
|    |    |    |    |    |    |__ Odoribacteraceae                     4       4       100.0
|    |    |    |    |    |__ Marinilabiliales                          10      10      100.0
|    |    |    |    |    |    |__ Marinilabiliaceae                    5       5       100.0
|    |    |    |    |    |    |__ Prolixibacteraceae                   3       3       100.0
|    |    |    |    |__ Cytophagia                                     70      46      65.7
|    |    |    |    |    |__ Cytophagales                              70      46      65.7
|    |    |    |    |    |    |__ Cyclobacteriaceae                    16      16      100.0
|    |    |    |    |    |    |    |__ Algoriphagus                     6       6       100.0
|    |    |    |    |    |    |__ Cytophagaceae                        22      11      50.0
|    |    |    |    |    |    |    |__ Dyadobacter                      4       3       75.0
|    |    |    |    |    |    |__ Hymenobacteraceae                    14      7       50.0
|    |    |    |    |    |    |    |__ Pontibacter                      4       3       75.0
|    |    |    |    |    |    |    |__ Hymenobacter                     8       3       37.5
|    |    |    |    |    |    |__ Flammeovirgaceae                     8       4       50.0
|    |    |    |    |    |    |__ Amoebophilaceae                      3       3       100.0
|    |    |    |    |__ Sphingobacteriia                               21      20      95.2
|    |    |    |    |    |__ Sphingobacteriales                        21      20      95.2
|    |    |    |    |    |    |__ Sphingobacteriaceae                  21      20      95.2
|    |    |    |    |    |    |    |__ Sphingobacterium                 8       8       100.0
|    |    |    |    |    |    |    |__ Pedobacter                       7       7       100.0
|    |    |    |    |__ Chitinophagia                                  12      12      100.0
|    |    |    |    |    |__ Chitinophagales                           12      12      100.0
|    |    |    |    |    |    |__ Chitinophagaceae                     12      12      100.0
|    |    |    |    |    |    |    |__ Flavihumibacter                  3       3       100.0
|    |    |    |__ Chlorobi                                            11      7       63.6
|    |    |    |    |__ Chlorobia                                      11      7       63.6
|    |    |    |    |    |__ Chlorobiales                              11      7       63.6
|    |    |    |    |    |    |__ Chlorobiaceae                        11      7       63.6
|    |    |    |    |    |    |    |__ Chlorobium/Pelodictyon group     8       6       75.0
|    |    |    |    |    |    |    |    |__ Chlorobium                  6       4       66.7
|    |__ Proteobacteria                                                1854    38      2.0
|    |    |__ Alphaproteobacteria                                      667     24      3.6
|    |    |    |__ Rickettsiales                                       42      16      38.1
|    |    |    |    |__ Rickettsiaceae                                 18      12      66.7
|    |    |    |    |    |__ Rickettsieae                              18      12      66.7
|    |    |    |    |    |    |__ Rickettsia                           15      10      66.7
|    |    |    |    |    |    |    |__ spotted fever group              12      8       66.7
|    |    |    |    |__ Anaplasmataceae                                21      4       19.0
```

```
|     |     |     |__ Pelagibacterales                                8      6     75.0
|     |     |     |     |__ Pelagibacteraceae                         8      6     75.0
|     |     |     |     |     |__ Candidatus Pelagibacter             5      3     60.0
|     |     |     |     |     |     |__ Candidatus Pelagibacter ubique 3     3     100.0
|     |     |     |     |     |__ unclassified Pelagibacteraceae      3      3     100.0
|     |     |__ Gammaproteobacteria                                   711    9     1.3
|     |     |     |__ Enterobacterales                                112    5     4.5
|     |     |     |     |__ Enterobacteriaceae                        41     3     7.3
|     |     |__ Betaproteobacteria                                    273    5     1.8
|     |     |     |__ unclassified Betaproteobacteria                 12     5     41.7
|     |     |     |     |__ Kinetoplastibacterium                     5      4     80.0
|     |__ Terrabacteria group                                         2228   38    1.7
|     |     |__ Cyanobacteria/Melainabacteria group                   127    32    25.2
|     |     |     |__ Cyanobacteria                                   127    32    25.2
|     |     |     |     |__ Nostocales                                31     13    41.9
|     |     |     |     |     |__ Nostocaceae                         10     9     90.0
|     |     |     |     |     |     |__ Nostoc                        4      4     100.0
|     |     |     |     |     |     |__ Anabaena                      3      3     100.0
|     |     |     |     |__ Oscillatoriophycideae                     31     9     29.0
|     |     |     |     |     |__ Oscillatoriales                     22     7     31.8
|     |     |     |     |__ Synechococcales                           55     8     14.5
|     |     |     |     |     |__ Synechococcaceae                    27     4     14.8
|     |     |     |     |     |     |__ Synechococcus                 24     4     16.7
|     |     |__ Tenericutes                                           116    4     3.4
|     |     |     |__ Mollicutes                                      115    4     3.5
|     |     |     |     |__ Mycoplasmatales                           73     4     5.5
|     |     |     |     |     |__ Mycoplasmataceae                    73     4     5.5
|     |     |     |     |     |     |__ Mycoplasma                    70     4     5.7
|     |__ PVC group                                                   52     7     13.5
|     |     |__ Verrucomicrobia                                       17     5     29.4
|     |__ Spirochaetes                                                60     2     3.3
|     |__ Acidobacteria                                               24     1     4.2
```

Group C

a: Total number of genomes in the taxon
b: Number of group C genomes in the taxon
c: Percentage of group C genomes in the taxon
d: Average percentage of first-genes-in-operon in the genomes of the taxon in prediction
e: Average percentage of predicted leaderless genes among FGIO
f: Average percentage of predicted leaderless genes among all genes

Colors Legend - Column c

- Individual colors have no specific meaning - colors are used to highlight members of the same "cluster", in particular, those clusters with a significant number of genomes that have been classified as class C.

Colors Legend - Column f

- Values between 10 and 20 percent
- Values between 20 and 30 percent
- Values between 30 and 40 percent
- Values between 40 and 60 percent
- "Higher level" groups that contain several of the above ranges have been left blank

	a	b	c	d	e	f
cellular organisms	5007	1028	20.5	70.4	36.7	31.4
|__ Bacteria	4769	1028	21.6	70.4	36.7	31.4
|__ Terrabacteria group	2228	877	39.4	71.2	37.3	31.7
|__ Actinobacteria	859	773	90.0	72.1	37.7	32.2
|__ Actinobacteria	807	764	94.7	72.1	37.6	32.1
|__ Micrococcales	211	198	93.8	69.6	39.5	33.2
|__ Microbacteriaceae	82	78	95.1	68.3	41.0	34.0
|__ Microbacterium	34	34	100.0	67.0	41.7	33.9
|__ Leifsonia	9	8	88.9	68.4	39.8	33.6
|__ Leucobacter	8	7	87.5	68.8	38.4	31.7
|__ Micrococcaceae	51	43	84.3	72.3	35.4	31.0
|__ Arthrobacter	19	16	84.2	71.8	36.3	31.7
|__ Kocuria	9	9	100.0	73.7	32.0	28.5
|__ Intrasporangiaceae	25	25	100.0	67.4	42.9	35.5
|__ Cellulomonadaceae	17	17	100.0	69.6	40.5	33.6
|__ Cellulomonas	14	14	100.0	69.6	40.3	33.4
|__ Brevibacteriaceae	7	7	100.0	69.8	35.2	29.5
|__ Brevibacterium	7	7	100.0	69.8	35.2	29.5
|__ Dermacoccaceae	6	6	100.0	70.5	42.4	35.2
|__ Corynebacteriales	202	197	97.5	70.4	41.0	35.0
|__ Corynebacteriaceae	72	70	97.2	70.7	40.8	35.3
|__ Corynebacterium	70	68	97.1	70.7	40.9	35.4
|__ Mycobacteriaceae	57	56	98.2	67.2	43.6	36.7
|__ Mycobacterium	56	55	98.2	67.2	43.7	36.8

|	|	|	|	|	|	|__ Nocardiaceae	43	41	95.3	74.2	36.9	32.0
|	|	|	|	|	|	| |__ Nocardia	30	28	93.3	75.8	36.3	31.6
|	|	|	|	|	|	| |__ Rhodococcus	12	12	100.0	71.0	37.8	32.6
|	|	|	|	|	|	|__ Gordoniaceae	19	19	100.0	70.8	42.5	35.8
|	|	|	|	|	|	| |__ Gordonia	19	19	100.0	70.8	42.5	35.8
|	|	|	|	|__ Streptomycetales	129	129	100.0	78.2	31.9	28.2
|	|	|	|	|	|__ Streptomycetaceae	129	129	100.0	78.2	31.9	28.2
|	|	|	|	|	|	|__ Streptomyces	117	117	100.0	78.0	32.0	28.2
|	|	|	|	|	|	|__ Kitasatospora	6	6	100.0	79.5	30.2	27.4
|	|	|	|	|__ Propionibacteriales	50	50	100.0	67.3	41.3	33.7
|	|	|	|	|	|__ Nocardioidaceae	32	32	100.0	66.7	42.7	35.0
|	|	|	|	|	|	|__ Nocardioides	18	18	100.0	66.2	43.0	35.2
|	|	|	|	|	|	|__ Aeromicrobium	6	6	100.0	66.0	44.0	35.9
|	|	|	|	|	|__ Propionibacteriaceae	18	18	100.0	68.5	38.7	31.3
|	|	|	|	|__ Pseudonocardiales	46	45	97.8	71.8	37.0	31.2
|	|	|	|	|	|__ Pseudonocardiaceae	46	45	97.8	71.8	37.0	31.2
|	|	|	|	|	|	|__ Saccharomonospora	9	9	100.0	72.7	37.4	31.3
|	|	|	|	|	|	|__ Amycolatopsis	7	7	100.0	70.7	36.2	30.1
|	|	|	|	|	|	|__ Pseudonocardia	7	7	100.0	70.6	38.0	32.3
|	|	|	|	|__ Actinomycetales	36	35	97.2	70.8	35.1	29.2
|	|	|	|	|	|__ Actinomycetaceae	36	35	97.2	70.8	35.1	29.2
|	|	|	|	|	|	|__ Actinomyces	23	22	95.7	71.0	36.3	30.1
|	|	|	|	|__ Streptosporangiales	34	32	94.1	76.6	32.3	28.2
|	|	|	|	|	|__ Nocardiopsaceae	18	16	88.9	78.6	30.4	26.9
|	|	|	|	|	|	|__ Nocardiopsis	15	15	100.0	78.4	30.5	26.9
|	|	|	|	|	|__ Streptosporangiaceae	9	9	100.0	74.4	35.4	30.7
|	|	|	|	|	|__ Thermomonosporaceae	7	7	100.0	74.7	32.6	28.2
|	|	|	|	|__ Bifidobacteriales	36	20	55.6	79.1	26.4	23.5
|	|	|	|	|	|__ Bifidobacteriaceae	36	20	55.6	79.1	26.4	23.5
|	|	|	|	|	|	|__ Bifidobacterium	30	14	46.7	80.1	26.7	23.8
|	|	|	|	|__ Micromonosporales	18	18	100.0	73.2	39.6	34.2
|	|	|	|	|	|__ Micromonosporaceae	18	18	100.0	73.2	39.6	34.2
|	|	|	|	|	|	|__ Actinoplanes	7	7	100.0	71.8	41.4	36.0
|	|	|	|	|__ Geodermatophilales	10	10	100.0	71.1	40.2	35.1
|	|	|	|	|	|__ Geodermatophilaceae	10	10	100.0	71.1	40.2	35.1
|	|	|	|	|__ Frankiales	11	8	72.7	75.6	34.8	31.6
|	|	|	|	|	|__ Frankiaceae	9	6	66.7	78.3	33.7	31.2
|	|	|	|	|	|	|__ Frankia	9	6	66.7	78.3	33.7	31.2
|	|	|__ Deinococcus-Thermus										
|	|	|	|__ Deinococci									
|	|	|	|	|__ Deinococcales	20	20	100.0	75.4	48.8	40.3
|	|	|	|	|	|__ Deinococcaceae	19	19	100.0	75.5	49.3	40.7
|	|	|	|	|	|	|__ Deinococcus	19	19	100.0	75.5	49.3	40.7
|	|	|	|	|__ Thermales	18	17	94.4	59.5	47.7	37.4
|	|	|	|	|	|__ Thermaceae	18	17	94.4	59.5	47.7	37.4
|	|	|	|	|	|	|__ Thermus	10	10	100.0	53.3	48.3	37.1
|	|	|	|	|	|	|__ Meiothermus	6	6	100.0	70.4	47.9	38.8

|	|	|__ Firmicutes	1064	36	3.4	64.1	19.4	17.1
|	|	|	|__ Bacilli	597	24	4.0	66.8	19.8	17.6
|	|	|	|	|__ Lactobacillales	255	24	9.4	66.8	19.8	17.6
|	|	|	|	|	|__ Streptococcaceae	41	19	46.3	66.8	18.8	16.9
|	|	|	|	|	|	|__ Streptococcus	39	18	46.2	66.5	18.5	16.6
|	|	|__ Tenericutes	116	30	25.9	61.5	32.7	26.5
|	|	|	|__ Mollicutes	115	30	26.1	61.5	32.7	26.5
|	|	|	|	|__ Mycoplasmatales	73	26	35.6	59.5	33.8	26.8
|	|	|	|	|	|__ Mycoplasmataceae	73	26	35.6	59.5	33.8	26.8
|	|	|	|	|	|	|__ Mycoplasma	70	26	37.1	59.5	33.8	26.8
|	|__ Proteobacteria	1854	104	5.6	66.5	32.2	29.9
|	|	|__ Gammaproteobacteria	711	34	4.8	70.2	32.0	29.5
|	|	|	|__ Xanthomonadales	62	18	29.0	69.8	35.7	32.4
|	|	|	|	|__ Xanthomonadaceae	48	18	37.5	69.8	35.7	32.4
|	|	|	|	|	|__ Lysobacter	12	7	58.3	72.7	35.2	32.2
|	|	|__ Alphaproteobacteria	667	34	5.1	67.9	33.2	31.4
|	|	|	|__ Sphingomonadales	105	13	12.4	68.1	32.6	30.3
|	|	|	|	|__ Sphingomonadaceae	83	10	12.0	68.3	32.5	30.4
|	|	|	|	|	|__ Sphingomonas	34	9	26.5	68.7	32.2	30.1
|	|	|__ delta/epsilon subdivisions	196	20	10.2	57.6	25.6	24.5
|	|	|	|__ Deltaproteobacteria	123	10	8.1	60.7	30.8	27.0
|	|	|	|	|__ Bdellovibrionales	8	7	87.5	65.6	30.0	26.9
|	|	|	|__ Epsilonproteobacteria	73	10	13.7	54.6	20.5	21.9
|	|	|	|	|__ Campylobacterales	67	9	13.4	54.5	20.5	22.2
|	|	|__ Betaproteobacteria	273	15	5.5	66.8	39.1	35.4
|	|	|	|__ Burkholderiales	166	15	9.0	66.8	39.1	35.4
|	|	|	|	|__ Comamonadaceae	50	8	16.0	67.0	40.0	35.7
|	|__ Spirochaetes	60	24	40.0	63.5	28.6	22.7
|	|	|__ Spirochaetia	60	24	40.0	63.5	28.6	22.7
|	|	|	|__ Spirochaetales	40	24	60.0	63.5	28.6	22.7
|	|	|	|	|__ Spirochaetaceae	28	21	75.0	63.7	29.2	22.8
|	|	|	|	|	|__ Treponema	17	12	70.6	64.3	29.7	23.6
|	|__ FCB group	455	8	1.8	70.2	56.9	52.5
|	|	|__ Bacteroidetes/Chlorobi group	450	7	1.6	70.3	59.9	55.7
|	|	|	|__ Bacteroidetes	436	7	1.6	70.3	59.9	55.7
|	|__ Deferribacteres	6	3	50.0	51.6	21.2	18.2
|	|__ Aquificae	14	3	21.4	45.4	39.4	29.9
|	|__ PVC group	52	2	3.8	74.8	50.8	46.1
|	|__ Acidobacteria	24	2	8.3	67.5	31.1	25.5
|	|__ Elusimicrobia	2	2	100.0	58.4	30.3	29.5
|	|__ unclassified Bacteria	5	1	20.0	70.9	41.4	35.3
|	|__ Fusobacteria	19	1	5.3	49.7	16.9	11.3
|	|__ Thermodesulfobacteria	6	1	16.7	53.0	33.0	30.0

Group D

a: Total number of genomes in the taxon
b: Number of group D genomes in the taxon
c: Percentage of group D genomes in the taxon
d: Average percentage of first-genes-in-operon in the genomes of the taxon in prediction
e: Average percentage of predicted leaderless genes among FGIO
f: Average percentage of predicted leaderless genes among all genes

Colors Legend - Column f

- Values between 10 and 25 percent
- Values between 25 and 40 percent
- Values between 40 and 55 percent
- Values between 55 and 70 percent
- "Higher level" groups that contain several of the above ranges have been left blank

	a	b	c	d	e	f
cellular organisms	5007	199	4.0	72.4	56.6	44.9
|__ Archaea	238	199	83.6	72.4	56.6	44.9
|__ Euryarchaeota	190	163	85.8	72.7	53.9	42.8
|__ Halobacteria	74	74	100.0	79.2	70.8	58.8
|__ Halobacteriales	26	26	100.0	78.1	70.5	57.9
|__ Haloarculaceae	9	9	100.0	78.4	73.4	60.0
|__ Halococcaceae	7	7	100.0	78.0	66.8	55.2
|__ Halococcus	7	7	100.0	78.0	66.8	55.2
|__ Halobacteriaceae	7	7	100.0	77.9	68.4	56.4
|__ Halobacterium	3	3	100.0	77.2	70.7	57.6
|__ unclassified Halobacteriales	3	3	100.0	78.3	75.0	61.4
|__ Natrialbales	25	25	100.0	81.5	71.1	60.5
|__ Natrialbaceae	25	25	100.0	81.5	71.1	60.5
|__ Halopiger	4	4	100.0	81.9	73.0	62.2
|__ Natronorubrum	3	3	100.0	81.2	71.9	61.2
|__ Natrinema	3	3	100.0	81.2	70.0	59.4
|__ Natronococcus	3	3	100.0	81.9	65.7	56.0
|__ Natrialba	3	3	100.0	81.8	70.8	60.7
|__ Haloferacales	23	23	100.0	77.8	71.0	57.9
|__ Halorubraceae	12	12	100.0	77.2	72.0	58.0
|__ Halorubrum	9	9	100.0	77.5	70.7	57.2
|__ Haloferacaceae	11	11	100.0	78.5	69.8	57.8
|__ Haloferax	5	5	100.0	77.7	71.7	58.8
|__ Methanomicrobia	42	40	95.2	72.5	36.4	28.8
|__ Methanosarcinales	26	24	92.3	76.6	29.6	24.9
|__ Methanosarcinaceae	22	20	90.9	78.6	28.2	24.4

| | | | | |__ Methanosarcina	13	12	92.3	83.0	27.4	24.5
| | | | | |__ Methanococcoides	3	3	100.0	71.9	29.2	24.1
| | | |__ Methanomicrobiales	14	14	100.0	65.8	47.0	35.1
| | | | |__ Methanomicrobiaceae	7	7	100.0	67.0	47.3	35.6
| | | | | |__ Methanoculleus	3	3	100.0	66.5	49.2	36.2
| | | |__ Methanoregulaceae	4	4	100.0	67.0	42.9	32.1
| | |__ Thermococci	21	21	100.0	63.0	30.3	21.4
| | | |__ Thermococcales	21	21	100.0	63.0	30.3	21.4
| | | | |__ Thermococcaceae	21	21	100.0	63.0	30.3	21.4
| | | | | |__ Thermococcus	14	14	100.0	64.2	31.7	22.8
| | | | | |__ Pyrococcus	5	5	100.0	59.1	28.3	19.0
| | |__ Thermoplasmata	11	11	100.0	65.7	63.2	45.8
| | | |__ Thermoplasmatales	6	6	100.0	67.5	72.3	52.7
| | | |__ Methanomassiliicoccales	5	5	100.0	63.5	52.3	37.6
| | | | |__ Methanomassiliicoccaceae	5	5	100.0	63.5	52.3	37.6
| | |__ Archaeoglobi	7	7	100.0	57.7	64.1	42.2
| | | |__ Archaeoglobales	7	7	100.0	57.7	64.1	42.2
| | | | |__ Archaeoglobaceae	7	7	100.0	57.7	64.1	42.2
| | | | | |__ Archaeoglobus	4	4	100.0	59.2	63.8	43.3
| | |__ Methanobacteria	18	5	27.8	66.3	26.2	21.3
| | | |__ Methanobacteriales	18	5	27.8	66.3	26.2	21.3
| | | | |__ Methanobacteriaceae	17	5	29.4	66.3	26.2	21.3
| | |__ Methanococci	15	3	20.0	63.6	12.5	11.9
| | | |__ Methanococcales	15	3	20.0	63.6	12.5	11.9
| | | | |__ Methanocaldococcaceae	8	3	37.5	63.6	12.5	11.9
| |__ TACK group	47	35	74.5	70.8	68.8	54.1
| | |__ Crenarchaeota	35	23	65.7	69.4	67.3	52.3
| | | |__ Thermoprotei	35	23	65.7	69.4	67.3	52.3
| | | | |__ Sulfolobales	11	11	100.0	66.6	69.0	52.0
| | | | | |__ Sulfolobaceae	11	11	100.0	66.6	69.0	52.0
| | | | | | |__ Sulfolobus	6	6	100.0	66.4	67.2	51.2
| | | | | | |__ Metallosphaera	3	3	100.0	66.7	68.3	50.5
| | | | |__ Thermoproteales	9	9	100.0	71.6	76.0	59.9
| | | | | |__ Thermoproteaceae	6	6	100.0	72.7	81.2	65.6
| | | | | |__ Thermofilaceae	3	3	100.0	69.3	65.6	48.4
| | | | | | |__ Thermofilum	3	3	100.0	69.3	65.6	48.4
| | |__ Thaumarchaeota	11	11	100.0	74.5	72.9	59.3
| | | |__ Nitrosopumilales	7	7	100.0	74.6	72.4	59.4
| | | | |__ Nitrosopumilaceae	7	7	100.0	74.6	72.4	59.4
| | | | | |__ Nitrosopumilus	5	5	100.0	74.8	73.1	59.7
| | | |__ unclassified Thaumarchaeota	3	3	100.0	72.2	78.1	61.7
| |__ unclassified Archaea	1	1	100.0	77.6	71.9	59.0

Group X

a: Total number of genomes in the taxon
b: Number of group X genomes in the taxon
c: Percentage of group X genomes in the taxon

	a	b	c
cellular organisms	5007	311	6.2
|__ Bacteria	4769	311	6.5
| |__ Proteobacteria	1854	142	7.7
| | |__ Betaproteobacteria	273	76	27.8
| | | |__ Burkholderiales	166	63	38.0
| | | | |__ Comamonadaceae	50	32	64.0
| | | | | |__ Acidovorax	7	6	85.7
| | | | | |__ Comamonas	7	5	71.4
| | | | |__ unclassified Burkholderiales	24	16	66.7
| | | | | |__ Burkholderiales Genera incertae sedis	21	14	66.7
| | | | | | |__ Thiomonas	4	3	75.0
| | | | |__ Burkholderiaceae	34	8	23.5
| | | | |__ Oxalobacteraceae	30	3	10.0
| | | | |__ Alcaligenaceae	23	3	13.0
| | | |__ unclassified Betaproteobacteria	12	5	41.7
| | | | |__ unclassified Betaproteobacteria (miscellaneous)	5	4	80.0
| | | |__ Methylophilales	14	4	28.6
| | | |__ Nitrosomonadales	9	4	44.4
| | | | |__ Nitrosomonadaceae	9	4	44.4
| | |__ Gammaproteobacteria	711	37	5.2
| | | |__ Xanthomonadales	62	9	14.5
| | | | |__ Xanthomonadaceae	48	8	16.7
| | | | | |__ Xanthomonas	6	3	50.0
| | | |__ Methylococcales	20	8	40.0
| | | | |__ Methylococcaceae	19	8	42.1
| | | | | |__ Methylomicrobium	3	3	100.0
| | | |__ unclassified Gammaproteobacteria	26	6	23.1
| | | |__ Enterobacterales	112	5	4.5
| | | | |__ Enterobacteriaceae	41	5	12.2
| | | | | |__ unclassified Enterobacteriaceae	13	5	38.5
| | | | | | |__ ant, tsetse, mealybug, aphid, etc. endosymbionts	10	5	50.0
| | | | | | | |__ ant endosymbionts	5	4	80.0
| | | | | | | | |__ Candidatus Blochmannia	5	4	80.0
| | | |__ Legionellales	38	3	7.9
| | |__ Alphaproteobacteria	667	26	3.9

```
|    |    |    |__ Rickettsiales                                       42      17     40.5
|    |    |    |    |__ Anaplasmataceae                                21      12     57.1
|    |    |    |    |    |__ Ehrlichia                                 6       6      100.0
|    |    |    |    |    |    |__ canis group                          5       5      100.0
|    |    |    |    |    |__ Wolbachieae                               8       4      50.0
|    |    |    |    |    |    |__ Wolbachia                            8       4      50.0
|    |    |    |    |__ Rickettsiaceae                                 18      5      27.8
|    |    |    |    |    |__ Rickettsieae                              18      5      27.8
|    |    |    |    |    |    |__ Rickettsia                           15      4      26.7
|    |    |    |    |    |    |    |__ spotted fever group             12      3      25.0
|    |    |    |__ Rhizobiales                                         225     3      1.3
|    |    |__ delta/epsilon subdivisions                               196     3      1.5
|    |    |    |__ Deltaproteobacteria                                 123     3      2.4
|    |    |    |    |__ Myxococcales                                   15      3      20.0
|    |    |    |    |    |__ Sorangiineae                              4       3      75.0
|    |__ Terrabacteria group                                           2228    107    4.8
|    |    |__ Cyanobacteria/Melainabacteria group                      127     90     70.9
|    |    |    |__ Cyanobacteria                                       127     90     70.9
|    |    |    |    |__ Synechococcales                                55      43     78.2
|    |    |    |    |    |__ Synechococcaceae                          27      20     74.1
|    |    |    |    |    |    |__ Synechococcus                        24      17     70.8
|    |    |    |    |    |__ Prochloraceae                             11      9      81.8
|    |    |    |    |    |    |__ Prochlorococcus                      11      9      81.8
|    |    |    |    |    |    |    |__ Prochlorococcus marinus         9       8      88.9
|    |    |    |    |    |__ Leptolyngbyaceae                          10      9      90.0
|    |    |    |    |    |    |__ Leptolyngbya                         8       7      87.5
|    |    |    |    |    |__ Pseudanabaenaceae                         4       3      75.0
|    |    |    |    |__ Oscillatoriophycideae                          31      22     71.0
|    |    |    |    |    |__ Oscillatoriales                           22      15     68.2
|    |    |    |    |    |    |__ Microcoleaceae                       6       5      83.3
|    |    |    |    |    |    |__ Oscillatoriaceae                     6       4      66.7
|    |    |    |    |    |    |__ Cyanothecaceae                       5       3      60.0
|    |    |    |    |    |    |    |__ Cyanothece                      5       3      60.0
|    |    |    |    |    |__ Chroococcales                             9       7      77.8
|    |    |    |    |    |    |__ Aphanothecaceae                      4       3      75.0
|    |    |    |    |__ Nostocales                                     31      18     58.1
|    |    |    |    |    |__ Rivulariaceae                             5       4      80.0
|    |    |    |    |    |    |__ Calothrix                            4       4      100.0
|    |    |    |    |    |__ Hapalosiphonaceae                         4       4      100.0
|    |    |    |    |    |__ Aphanizomenonaceae                        4       4      100.0
|    |    |    |    |    |__ Tolypothrichaceae                         4       3      75.0
|    |    |    |    |    |    |__ Tolypothrix                          4       3      75.0
|    |    |    |    |__ Pleurocapsales                                 5       5      100.0
```

```
|     |     |     |     |     |__ Hyellaceae                          3       3      100.0
|     |     |__ Tenericutes                                           116     10     8.6
|     |     |     |__ Mollicutes                                      115     10     8.7
|     |     |     |     |__ Mycoplasmatales                           73      10     13.7
|     |     |     |     |     |__ Mycoplasmataceae                    73      10     13.7
|     |     |     |     |     |     |__ Mycoplasma                    70      10     14.3
|     |     |__ Actinobacteria                                        859     7      0.8
|     |     |     |__ Actinobacteria                                  807     4      0.5
|     |     |     |__ Thermoleophilia                                 8       3      37.5
|     |     |     |     |__ Solirubrobacterales                       8       3      37.5
|     |__ FCB group                                                   455     35     7.7
|     |     |__ Bacteroidetes/Chlorobi group                          450     34     7.6
|     |     |     |__ Bacteroidetes                                   436     29     6.7
|     |     |     |     |__ Cytophagia                                70      24     34.3
|     |     |     |     |     |__ Cytophagales                        70      24     34.3
|     |     |     |     |     |     |__ Cytophagaceae                 22      11     50.0
|     |     |     |     |     |     |     |__ Spirosoma               3       3      100.0
|     |     |     |     |     |     |__ Hymenobacteraceae             14      7      50.0
|     |     |     |     |     |     |     |__ Hymenobacter            8       5      62.5
|     |     |     |     |     |     |__ Flammeovirgaceae              8       4      50.0
|     |     |     |     |__ Bacteroidia                               137     3      2.2
|     |     |     |     |     |__ Bacteroidales                       127     3      2.4
|     |     |     |__ Chlorobi                                        11      4      36.4
|     |     |     |     |__ Chlorobia                                 11      4      36.4
|     |     |     |     |     |__ Chlorobiales                        11      4      36.4
|     |     |     |     |     |     |__ Chlorobiaceae                 11      4      36.4
|     |__ PVC group                                                   52      15     28.8
|     |     |__ Verrucomicrobia                                       17      12     70.6
|     |     |     |__ Verrucomicrobiae                                7       5      71.4
|     |     |     |     |__ Verrucomicrobiales                        7       5      71.4
|     |     |     |__ Opitutae                                        4       3      75.0
|     |     |__ Planctomycetes                                        16      3      18.8
|     |     |     |__ Planctomycetia                                  15      3      20.0
|     |     |     |     |__ Planctomycetales                          13      3      23.1
|     |     |     |     |     |__ Planctomycetaceae                   9       3      33.3
|     |__ Spirochaetes                                                60      8      13.3
|     |     |__ Spirochaetia                                          60      8      13.3
|     |     |     |__ Leptospirales                                   16      8      50.0
|     |     |     |     |__ Leptospiraceae                            16      8      50.0
|     |     |     |     |     |__ Leptospira                          14      6      42.9
|     |__ Acidobacteria                                               24      4      16.7
```

REFERENCES

[1] J Shine and L Dalgarno. “The 3’-terminal sequence of Escherichia coli 16S ribosomal RNA: complementarity to nonsense triplets and ribosome binding sites”. In: *Proceedings of the National Academy of Sciences of the United States of America* 71.4 (Apr. 1974), pp. 1342–1346.

[2] Lars B Scharff et al. “Local absence of secondary structure permits translation of mRNAs that lack ribosome-binding sites.” In: *PLoS genetics* 7.6 (June 2011), e1002155.

[3] So Nakagawa, Yoshihito Niimura, and Takashi Gojobori. “Comparative genomic analysis of translation initiation mechanisms for genes lacking the Shine-Dalgarno sequence in prokaryotes”. In: *Nucleic acids research* 45.7 (Apr. 2017), pp. 3922–3931.

[4] Xiaobin Zheng et al. “Leaderless genes in bacteria: clue to the evolution of translation initiation mechanisms in prokaryotes”. In: *BMC Genomics* 12.1 (2011), p. 361.

[5] Damilola Omotajo et al. “Distribution and diversity of ribosome binding sites in prokaryotic genomes.” In: *BMC genomics* 16.1 (Aug. 2015), p. 604.

[6] Julia Babski et al. “Genome-wide identification of transcriptional start sites in the haloarchaeon Haloferax volcanii based on differential RNA-Seq (dRNA-Seq)”. In: *BMC Genomics* 17.1 (2016), p. 629.

[7] Teresa Cortes et al. “Genome-wide mapping of transcriptional start sites defines an extensive leaderless transcriptome in Mycobacterium tuberculosis”. In: *Cell reports* 5.4 (Nov. 2013), pp. 1121–1131.

[8] Cynthia M Sharma et al. “The primary transcriptome of the major human pathogen Helicobacter pylori”. In: *Nature* 464.7286 (2010), pp. 250–255.

[9] Pierre Nicolas et al. “Condition-Dependent Transcriptome Reveals High-Level Regulatory Architecture in ¡em¿Bacillus subtilis¡/em¿”. In: *Science* 335.6072 (Mar. 2012), 1103 LP –1106.

[10] Carsten Kröger et al. “An Infection-Relevant Transcriptomic Compendium for ¡em¿Salmonella enterica¡/em¿ Serovar Typhimurium”. In: *Cell Host & Microbe* 14.6 (Dec. 2013), pp. 683–695.

[11] Sandra Wiegand et al. “RNA-Seq of Bacillus licheniformis: active regulatory RNA features expressed within a productive fermentation”. In: *BMC genomics* 14 (Oct. 2013), p. 667.

[12] Gaurav Dugar et al. “High-Resolution Transcriptome Maps Reveal Strain-Specific Regulatory Features of Multiple Campylobacter jejuni Isolates”. In: *PLOS Genetics* 9.5 (May 2013), e1003495.

[13] Joseph McLaughlin et al. “Propionibacterium acnes and Acne Vulgaris: New Insights from the Integration of Population Genetic, Multi-Omic, Biochemical and Host-Microbe Studies.” In: *Microorganisms* 7.5 (May 2019).

[14] Wenjun Shao et al. “Conservation of transcription start sites within genes across a bacterial genus”. In: *mBio* 5.4 (July 2014), e01398.

[15] Maureen K Thomason et al. “Global transcriptional start site mapping using differential RNA sequencing reveals novel antisense RNAs in Escherichia coli”. In: *Journal of bacteriology* 197.1 (Jan. 2015), pp. 18–28.

[16] Dominik Jäger et al. “Deep sequencing analysis of the Methanosarcina mazei Gö1 transcriptome in response to nitrogen availability”. In: *Proceedings of the National Academy of Sciences of the United States of America* 106.51 (Dec. 2009), pp. 21878–21882.

[17] Claire Toffano-Nioche et al. “RNA at 92 °C: the non-coding transcriptome of the hyperthermophilic archaeon Pyrococcus abyssi”. In: *RNA biology* 10.7 (July 2013), pp. 1211–1220.

[18] Dominik Jäger et al. “Primary transcriptome map of the hyperthermophilic archaeon Thermococcus kodakarensis”. In: *BMC genomics* 15.1 (Aug. 2014), p. 684.

[19] Jie Li et al. “Global mapping transcriptional start sites revealed both transcriptional and post-transcriptional regulation of cold adaptation in the methanogenic archaeon Methanolobus psychrophilus”. In: *Scientific reports* 5 (Mar. 2015), p. 9209.

[20] Suhyung Cho et al. “Genome-wide primary transcriptome analysis of H(2)-producing archaeon Thermococcus onnurineus NA1”. In: *Scientific reports* 7 (Feb. 2017), p. 43044.

[21] Katharina Pfeifer-Sancar et al. “Comprehensive analysis of the Corynebacterium glutamicum transcriptome using an improved RNAseq technique”. In: *BMC genomics* 14 (Dec. 2013), p. 888.

[22] Arjan de Groot et al. “RNA sequencing and proteogenomics reveal the importance of leaderless mRNAs in the radiation-tolerant bacterium Deinococcus deserti”. In: *Genome biology and evolution* 6.4 (Apr. 2014), pp. 932–948.

[23] David A Romero et al. “A comparison of key aspects of gene regulation in Streptomyces coelicolor and Escherichia coli using nucleotide-resolution transcription

maps produced in parallel by global and differential RNA sequencing". In: *Molecular microbiology* 94.5 (Sept. 2014), pp. 963–987.

[24] Scarlet S Shell et al. "Leaderless Transcripts and Small Proteins Are Common Features of the Mycobacterial Translational Landscape". In: *PLoS genetics* 11.11 (Nov. 2015), e1005641–e1005641.

[25] Tie Koide et al. "Prevalence of transcription promoters within archaeal operons and coding sequences". In: *Molecular systems biology* 5 (2009), p. 285.

[26] Omri Wurtzel et al. "A single-base resolution map of an archaeal transcriptome". In: *Genome research* 20.1 (Jan. 2010), pp. 133–141.

[27] D F Reim and D W Speicher. "N-terminal sequence analysis of proteins and peptides". In: *Current protocols in protein science* Chapter 11 (May 2001), Unit–11.10.

[28] Scott Mann and Yi-Ping Phoebe Chen. "Bacterial genomic G+C composition-eliciting environmental adaptation". In: *Genomics* 95.1 (2010), pp. 7–15.

[29] Kenneth J Locey and Jay T Lennon. "Scaling laws predict global microbial diversity". In: *Proceedings of the National Academy of Sciences* 113.21 (May 2016), 5970 LP –5975.

[30] Brian Søgaard Laursen et al. "Initiation of protein synthesis in bacteria". In: *Microbiology and molecular biology reviews : MMBR* 69.1 (Mar. 2005), pp. 101–123.

[31] Herman A. De Boer and Anna S. Hui. "Sequences within ribosome binding site affecting messenger RNA translatability and method to direct ribosomes to single messenger RNA species". In: *Methods in Enzymology* 185 (Jan. 1990), pp. 103–114.

[32] G D Stormo, T D Schneider, and L M Gold. "Characterization of translational initiation sites in E. coli". In: *Nucleic acids research* 10.9 (May 1982), pp. 2971–2996.

[33] Isabella Moll et al. "Leaderless mRNAs in bacteria: surprises in ribosomal recruitment and translational control." In: *Molecular microbiology* 43.1 (Jan. 2002), pp. 239–246.

[34] Arthur L Delcher et al. "Identifying bacterial genes and endosymbiont DNA with Glimmer". In: *Bioinformatics (Oxford, England)* 23.6 (Mar. 2007), pp. 673–679.

[35] J Besemer, A Lomsadze, and M Borodovsky. "GeneMarkS: a self-training method for prediction of gene starts in microbial genomes. Implications for finding sequence motifs in regulatory regions". In: *Nucleic acids research* 29.12 (June 2001), pp. 2607–2618.

[36] Doug Hyatt et al. “Prodigal: prokaryotic gene recognition and translation initiation site identification”. In: *BMC bioinformatics* 11 (Mar. 2010), p. 119.

[37] M Borodovsky et al. “Detection of new genes in a bacterial genome using Markov models for three gene classes”. In: *Nucleic acids research* 23.17 (Sept. 1995), pp. 3554–3562.

[38] Wenhan Zhu, Alexandre Lomsadze, and Mark Borodovsky. “Ab initio gene identification in metagenomic sequences”. In: *Nucleic acids research* 38.12 (July 2010), e132–e132.

[39] William Thompson, Eric C Rouchka, and Charles E Lawrence. “Gibbs Recursive Sampler: finding transcription factor binding sites”. In: *Nucleic acids research* 31.13 (July 2003), pp. 3580–3585.

[40] S F Altschul et al. “Basic local alignment search tool.” In: *Journal of molecular biology* 215.3 (Oct. 1990), pp. 403–410.

[41] Dennis A Benson et al. “GenBank.” In: *Nucleic acids research* 43.Database issue (Jan. 2015), pp. D30–5.

[42] *ELPH: Estimated Locations of Pattern Hits*.

[43] Luis Javier Rodríguez and Inés Torres. “Comparative Study of the Baum-Welch and Viterbi Training Algorithms Applied to Read and Spontaneous Speech Recognition BT - Pattern Recognition and Image Analysis”. In: ed. by Francisco José Perales et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 847–857. ISBN: 978-3-540-44871-6.

[44] Takashi Sazuka, Minoru Yamaguchi, and Osamu Ohara. “Cyano2Dbase updated: linkage of 234 protein spots to corresponding genes through N-terminal microsequencing.” In: 1999.

[45] K E Rudd. “EcoGene: a genome sequence database for Escherichia coli K-12”. In: *Nucleic acids research* 28.1 (Jan. 2000), pp. 60–64.

[46] Jocelyne M. Lew et al. “TubercuList – 10 years after”. In: *Tuberculosis* 91.1 (Jan. 2011), pp. 1–7.

[47] Syuji Yamazaki et al. “Proteome Analysis of an Aerobic Hyperthermophilic Crenarchaeon, ¡em¿Aeropyrum pernix¡/em¿ K1”. In: *Molecular & Cellular Proteomics* 5.5 (May 2006), 811 LP –823.

[48] Michalis Aivaliotis et al. “Large-Scale Identification of N-Terminal Peptides in the Halophilic Archaea Halobacterium salinarum and Natronomonas pharaonis”. In: *Journal of Proteome Research* 6.6 (June 2007), pp. 2195–2204.

[49] Tatiana Tatusova et al. “RefSeq microbial genomes database: new representation and annotation strategy”. In: *Nucleic acids research* 42.Database issue (Jan. 2014), pp. D553–D559.

[50] Jan Mitschke et al. “An experimentally anchored map of transcriptional start sites in the model cyanobacterium Synechocystis sp. PCC6803”. In: *Proceedings of the National Academy of Sciences of the United States of America* 108.5 (Feb. 2011), pp. 2124–2129.

[51] Claudio O Gualerzi et al. “Expanded microbial genome coverage and improved protein family annotation in the COG database”. In: *Nucleic acids research* 5.7 (Jan. 2013), pp. 123–133.

[52] C. S. Shean and M. E. Gottesman. “Translation of the prophage λ cl transcript”. In: *Cell* 70.3 (1992), pp. 513–522.

[53] D Barrick et al. “Quantitative analysis of ribosome binding sites in E.coli”. In: *Nucleic acids research* 22.7 (Apr. 1994), pp. 1287–1295.

[54] A Resch et al. “Downstream box-anti-downstream box interactions are dispensable for translation initiation of leaderless mRNAs”. In: *The EMBO journal* 15.17 (Sept. 1996), pp. 4740–4748.

[55] Claudio O Gualerzi and Cynthia L Pon. “Initiation of mRNA translation in bacteria: structural and dynamic aspects”. In: *Cellular and molecular life sciences : CMLS* 72.22 (Nov. 2015), pp. 4341–4367.

[56] Udo Wegmann, Nikki Horn, and Simon R Carding. “Defining the Bacteroides Ribosomal Binding Site”. In: *Applied and Environmental Microbiology* 79.6 (Mar. 2013), 1980 LP –1989.

[57] Kyungtaek Lim, Yoshikazu Furuta, and Ichizo Kobayashi. “Large variations in bacterial ribosomal RNA genes”. In: *Molecular biology and evolution* 29.10 (Oct. 2012), pp. 2937–2948.

[58] Alexandre Lomsadze et al. “Modeling leaderless transcription and atypical genes results in more accurate gene prediction in prokaryotes”. In: *Genome Research* 28.7 (2018), pp. 1079–1089.

[59] Tatiana Tatusova et al. “NCBI prokaryotic genome annotation pipeline”. In: *Nucleic acids research* 44.14 (Aug. 2016), pp. 6614–6624.

[60] Michael E Wall et al. “Genome majority vote improves gene predictions”. In: *PLoS computational biology* 7.11 (Nov. 2011), e1002284–e1002284.

[61] Benjamin Buchfink, Chao Xie, and Daniel H Huson. “Fast and sensitive protein alignment using DIAMOND”. In: *Nature Methods* 12.1 (2015), pp. 59–60.

[62] Motoo Kimura. “A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences”. In: *Journal of Molecular Evolution* 16.2 (1980), pp. 111–120.

[63] Fabian Sievers and Desmond G Higgins. “Clustal Omega for making accurate alignments of many protein sequences”. In: *Protein Science* 27.1 (Jan. 2018), pp. 135–145.

[64] Madeleine Huber et al. “Translational coupling via termination-reinitiation in archaea and bacteria”. In: *Nature Communications* 10.1 (Dec. 2019), pp. 1–11.

[65] Jindan Zhou and Kenneth E Rudd. “EcoGene 3.0.” In: *Nucleic acids research* 41.Database issue (Jan. 2013), pp. D613–24.

[66] Céline Bland et al. “N-Terminal-oriented proteogenomics of the marine bacterium roseobacter denitrificans Och114 using N-Succinimidyloxycarbonylmethyl)tris(2,4,6-trimethoxyphenyl)phosphonium bromide (TMPP) labeling and diagonal chromatography.” In: *Molecular & cellular proteomics : MCP* 13.5 (May 2014), pp. 1369–1381.

[67] *The New Science of Metagenomics: Revealing the Secrets of Our Microbial Planet*. 2007. ISBN: 978-0-309-10676-4.

[68] Hui-Qi Zhou et al. “Analysis of the Relationship between Genomic GC Content and Patterns of Base Usage, Codon Usage and Amino Acid Usage in Prokaryotes: Similar GC Content Adopts Similar Compositional Frequencies Regardless of the Phylogenetic Lineages”. In: *PLOS ONE* 9.9 (Sept. 2014), e107319.

[69] Doug Hyatt et al. “Gene and translation initiation site prediction in metagenomic sequences.” In: *Bioinformatics (Oxford, England)* 28.17 (Sept. 2012), pp. 2223–2230.

[70] Mina Rho, Haixu Tang, and Yuzhen Ye. “FragGeneScan: predicting genes in short and error-prone reads”. In: *Nucleic Acids Research* 38.20 (Nov. 2010), e191–e191.

[71] Hideki Noguchi, Takeaki Taniguchi, and Takehiko Itoh. “MetaGeneAnnotator: Detecting Species-Specific Patterns of Ribosomal Binding Site for Precise Gene Prediction in Anonymous Prokaryotic and Phage Genomes”. In: *DNA Research* 15.6 (Oct. 2008), pp. 387–396.

[72] Axel Tiessen, Paulino Pérez-Rodríguez, and Luis José Delaye-Arredondo. “Mathematical modeling and comparison of protein size distribution in different plant, animal, fungal and microbial species reveals a negative correlation between protein size and protein number, thus providing insight into the evolution of proteomes”. In: *BMC Research Notes* 5.1 (2012), p. 85.

[73] S D Bell and S P Jackson. “Transcription and translation in Archaea: a mosaic of eukaryal and bacterial features.” In: *Trends in microbiology* 6.6 (June 1998), pp. 222–228.

[74] Alexandra M Gehring, Julie E Walker, and Thomas J Santangelo. “Transcription Regulation in Archaea”. In: *Journal of Bacteriology* 198.14 (July 2016). Ed. by W Margolin, 1906 LP –1917.